K Nearest Neighbor Classifiers for prediction on stream data

Project report submitted in partial fulfillment of the requirement for the degree of Bachelor of Technology

in

Information Technology

By

Aashish(141447)

Under the supervision of

Dr. Pardeep Kumar

to

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology Waknaghat, Solan-173234, Himachal Pradesh

Candidate's Declaration

I here by declare that the work presented in this report entitled " K Nearest Neigbhor classifier for prediction on stream data " in fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Information Technology** submitted in the department of Computer Science & Engineering and Information Technology**,** Jaypee University of Information Technology Waknaghat is an authentic record of our own work carried out over a period from August 2017 to May 2018 under the supervision of **Dr. Pardeep kumar (** Associate Professor (Senior Grade) in the Department of Computer Science & Engineering at Jaypee University of Information Technology (JUIT)**)**.

The matter embodied in the report has not been submitted for the award of any other degree or diploma.

Aashish, 141447

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

Supervisor Name: Dr. Pardeep Kumar
Designation: Associate Professor
Department name: Computer Science &Engineering
Dated:12-5-2018

Acknowledgement

I take this opportunity to express our profound gratitude and deep regards to our guide Dr. Pardeep kumar for his exemplary guidance, monitoring and constant encouragement throughout the course of this project. The blessing, help and guidance given by his time to time shall carry me a long way in the journey of life on which I am about to embark.

The in-time facilities provided by the Computer Science department throughout the project development are also equally acknowledgeable.

At the end I would like to express our sincere thanks to all my friends and others who helped us directly or indirectly during this project work.

Date: 12-8-2018 **Aashish (141447)**

ABSTRACT

KNN is an extensively used classification algorithm owing to its simplicity, ease of implementation and effectiveness. It is one of the top ten data mining algorithms, has been widely applied in various fields. KNN has few shortcomings affecting its acc of classification. It has large memory requirements as well as high time complexity. Several techniques have been proposed to improve these shortcomings in literature. In this paper, we have first reviewed some improvements made in KNN algorithm. Then, we have proposed our novel improved algorithm. It is a combination of dynamic selected, attribute weighted and distance weighted techniques like WKNN and DC for KNN and DBSCAN respectively. We have experimentally tested our proposed algorithm in R Studio, using a standard UCI dataset-Iris. The accuracy of our algorithm is improved with a blend of classification and clustering techniques. Experimental results have proved that our proposed algorithm performs better than conventional KNN algorithm.

Table Content

List of Figures

List of Graph

List of Tables

Chapter 1 – Introduction

1.1 Introduction

1.1.1 Need:

1. Optimize website profitability by making appropriate offers to each visitor.
2. Predict customer defections.
3. Identify unusual behavior.
4. Identify unexpected shopping patterns in supermarkets.
5. To analyzing large amounts of data in an effort to find correlations.
6. To obtain needful knowledge from our noisy data.
7. Data Mining can be applied anywhere in organization where you are interested in identify and predict output.

1.1.2 Data Mining:

Data mining is the extraction of interesting models or knowledge from a large amount of data. The data mining technique is used to predict group membership of data instances and these are used in forecasting future data sets.

Alternative names: - Knowledge discovery (mining) in databases (KDD), knowledge extraction, data / pattern analysis, data archeology, data dredging, information gathering, business intelligence, etc.

Data mining is not a: -

- query processing
- Expert system or statistical programs.

Data mining has attracted great attention in the information industry and society as a whole in recent years, due to the widespread availability of huge amounts of data and the imminent need to turn this data into useful information and knowledge. The information and knowledge achieve can be used for applications ranging from fraud detection, market analysis, and customer loyalty, production control and scientific exploration.

Data mining can be seen as a result of the natural evolution of information technology. The database systems industry has seen an evolutionary path in the development of the following on functionalities (Figure 1): data collection , creation, management respectively (including data storage and retrieval, and database transaction processing), and advanced data analysis (involving data warehousing and mining).

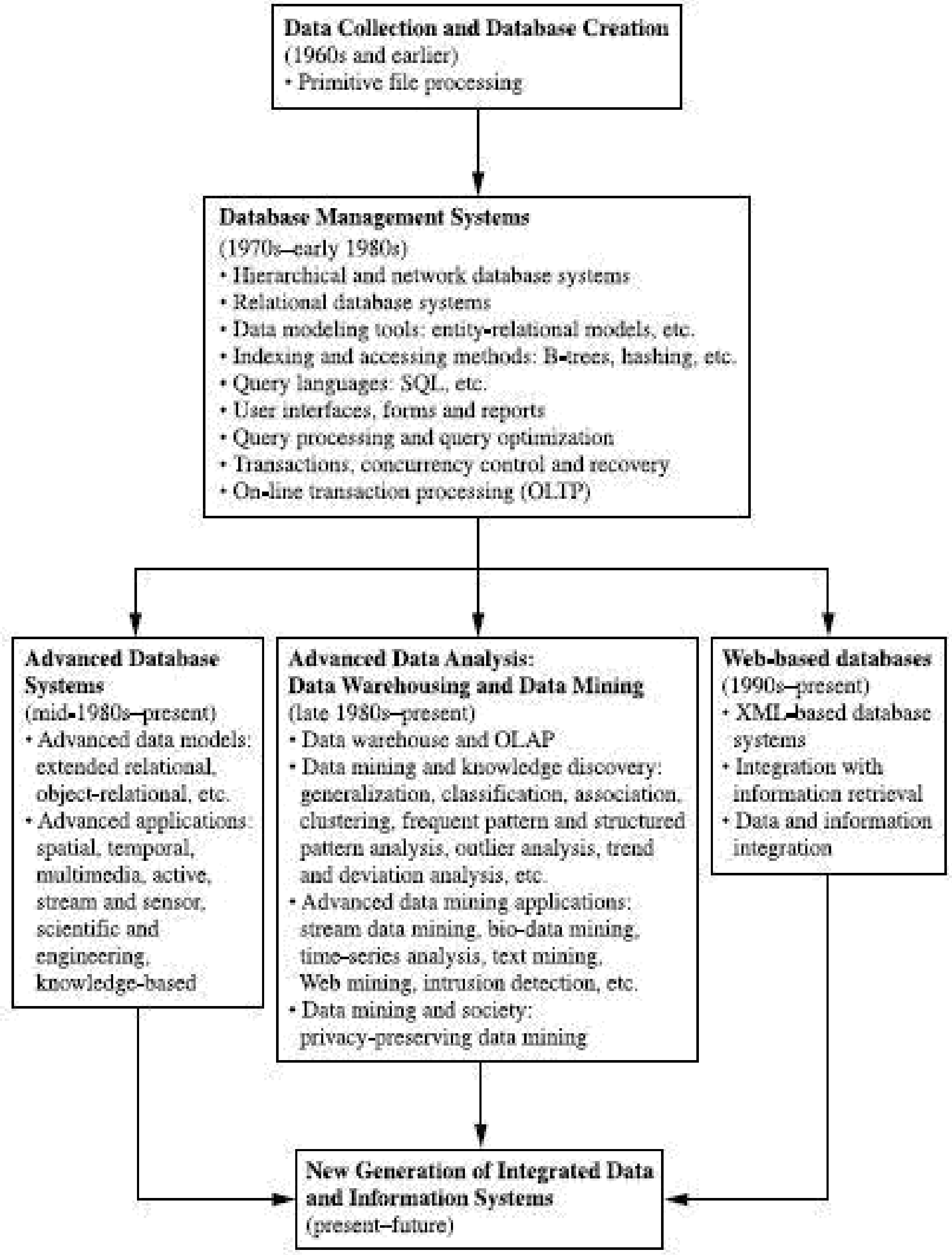

Data Collection and Database Creation
(1960s and earlier)
• Primitive file processing
Database Management Systems
(1970s—early 1980s)
• Hierarchical and network database systems
• Relational database systems
• Data modeling tools: entity-relational models, etc.
• Indexing and accessing methods: B-trees, hashing, etc.
• Query languages: SQL, etc.
• User interfaces, forms and reports
• Query processing and query optimization
• Transactions, concurrency control and recovery
• On-line transaction processing (OLTP)
Advanced Database Systems
(mid-1980s—present)
• Advanced data models: extended relational, object-relational, etc.
• Advanced applications: spatial, temporal, multimedia, active, stream and sensor, scientific and engineering, knowledge-based
Advanced Data Analysis: Data Warehousing and Data Mining
(late 1980s—present)
• Data warehouse and OLAP
• Data mining and knowledge discovery: generalization, classification, association, clustering, frequent pattern and structured pattern analysis, outlier analysis, trend and deviation analysis, etc.
• Advanced data mining applications: stream data mining, bio-data mining, time-series analysis, text mining, Web mining, intrusion detection, etc.
• Data mining and society: privacy-preserving data mining
Web-based databases
(1990s—present)
• XML-based database systems
• Integration with information retrieval
• Data and information integration
New Generation of Integrated Data and Information Systems
(present–future)

Figure 1: Data Collection

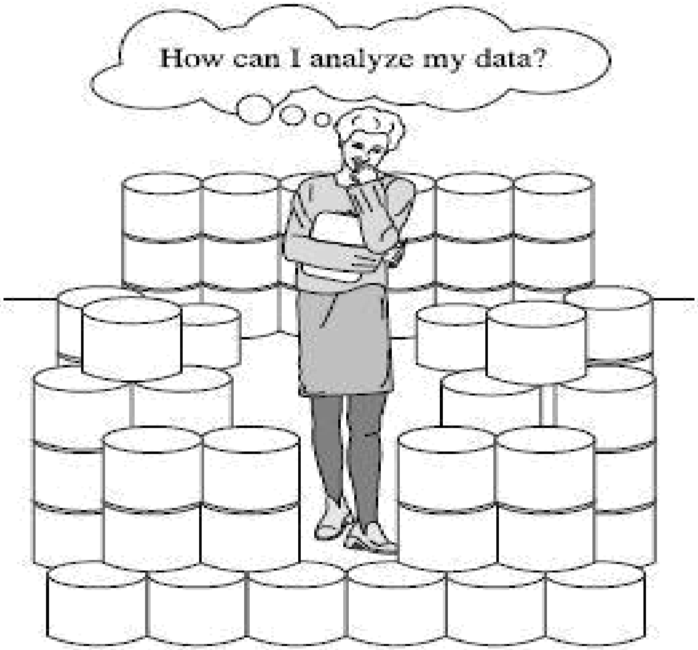


Figure 2

Figure 3

For example, the initial development of data collection and database creation mechanisms served as a prerequisite for the further development of effective data storage and retrieval mechanisms and the processing of queries and transactions. With numerous database systems that offer query and transaction processing as a common practice, advanced data analysis has naturally become the next goal. Since the 1960s, the database and information technology have evolved systematically from elementary file processing systems to sophisticated and all- powerful database systems. Research and development in database systems since the '70s have gone from the first ordered and network database systems to the development of relational database systems. Database technology since the mid-1980s has been characterized by the suitable adoption of relational technology and by an progress in research and development in new and all-powerful database systems. These promote the development of advanced data models as models of expansive, object-oriented, object-related and deductive relationships. Application-oriented database systems have flourished, including spatial, temporal, multimedia, active, continuous and sensor databases, scientific databases, knowledge bases and office information. The problems related to the heterogeneousness ,distribution, and swap of data have been studied in depth.

Now the data can be stored in different types of databases and information repositories. A data vault architecture that has come out is the data warehouse, a repository of numerious different types of data sources organized into a unified schema in a single site to facilitate management decision making.

1.1.3 Data Mining Definition:

Data mining technique is used to predict group membership to data instances and these are utilized in prediction of future data sets, interesting patterns. Easily stated, data mining refers to extricate or “mining” knowledge from huge volume of data. The is veritably a misname. And remember that the mining of gold from rocks and sand is revered to as gold mining rather than rock and mining. Just like that, data mining should have been more conveniently named knowledge mining from statistics, which is unfortunately somewhat long. Knowledge mining, a lower term, may not imitate the feature on mining from huge volume of data. Nonetheless, mining is a clear term characterizing the process that finds a small-scale set of inestimatable hunk from a great deal of impure material. Thus, such a misnomer that sustain both “information” and “mining” became a suitable choice. Many other terms carry a similar or slightly different meaning to data mining, such as knowledge mining from data, knowledge extraction, data/pattern analysis, data archaeology, and data dredging.

The steady and amazing progress of computer hardware technology in the past three decades has led to large supplies of powerful and affordable computers, data collection equipment, and storage media. This technology provides a great boost to the database and information industry, and makes a huge number of databases and information repositories available for transaction management, information retrieval, and data analysis.

1.2 Objective:

To predict the value of a target field in input data based on the values of its nearest neighbors, or to simply find which are the nearest neighbors for a particular case of interest.

1.3 Methodology:

K means, K Nearest Neighbors, Regression Techniques etc.

Chapter 2 Literature Survey

2.1 Knowledge Discovery in Database:

The KDD process is interchangeable and iterative, involving numerous steps with many decisions made by the user. Brachman and Anand (1996) give a practical view of the KDD process, focusing the interchangeable nature of the process.

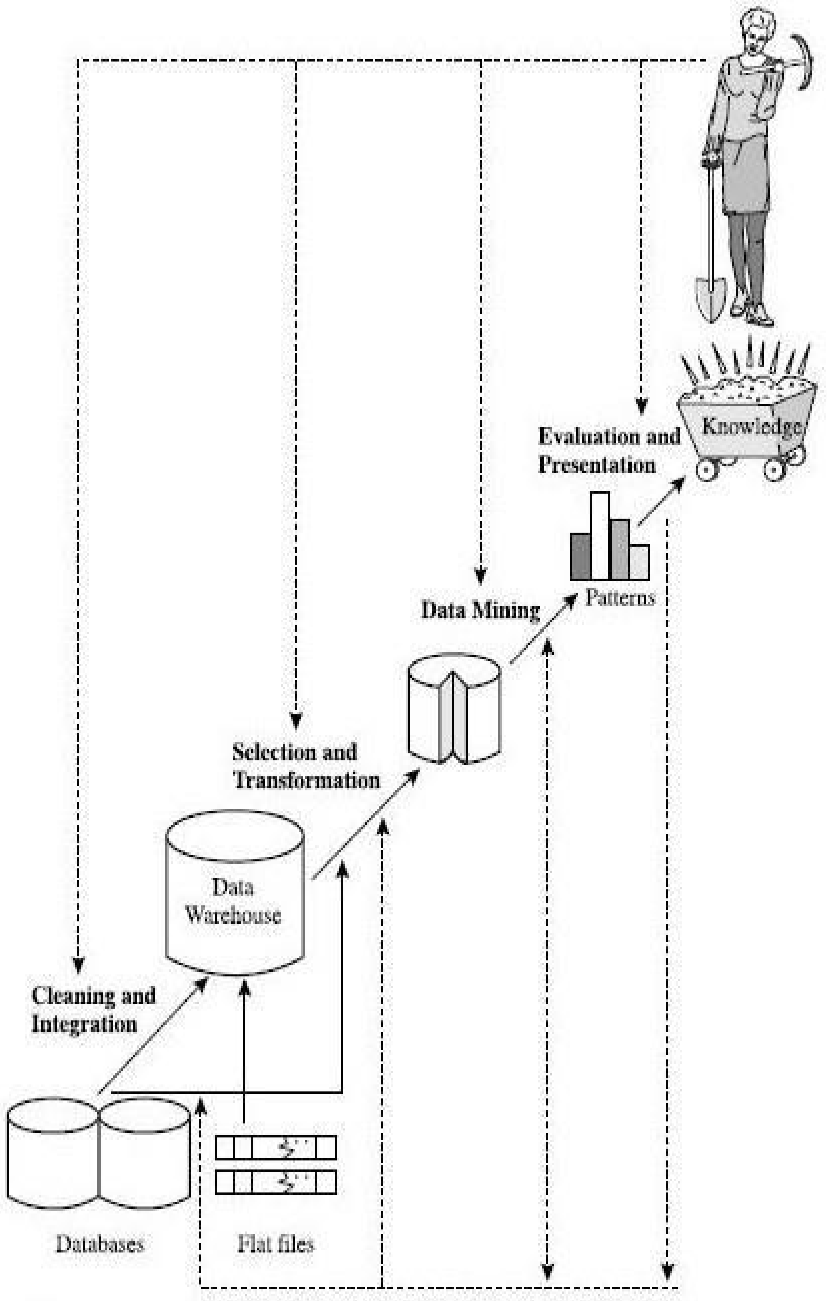


Figure 4: KDD Process

1. Data cleaning:-to extract noise and unpredictable data
2. Data integration:-where numerous data supplies may be combined
3. Data selection:-where data related to the analysis task are extracted from the database
4. Data transformation:- where data are reconstruct or consolidated into forms convenient for mining by performing condensed or aggregation operations, for instance
5. Data mining:-an needful process where intelligent technic are applied in order to retrieve data patterns
6. Pattern evaluation:-to identify the truly impressive patterns that show knowledge based on some impressive measures
7. Knowledge presentation:-where visualize and knowledge delegation techniques are used to represent the mined knowledge to the user

2.2 Methods:

2.2.1 Regression Method

Regression analysis is a statistical technique that is most often used for numeric estimation, although other methods exist as well. Estimation also encompasses the assimilation of distribution trends based on the available data.

Linear Regression: Regression is learning a function that maps a data item to a real-valued estimation variable. Regression applications are numerous, for example, estimating the volume of biomass present in a forest given unexpectedly sensed microwave measurements, estimating the expectation that a patient will survive in the given results of a set of diagnostic tests and estimating consumer exaction for a new product as a function of advertising consumption, and estimating time series where the input variables can be time-lagged versions of the estimation variable. Figure 5a show the result of simple linear regression where total debt is matched as a linear function of income: The fit is underprivileged because only a feeble correlation exists between the two variables.

Non Linear Regression: These methods consist of a network of techniques for estimation that fit linear and nonlinear combinations of basis functions to combinations of the input variables. Examples include feed forward neural networks, adjusting sapling methods, and prolongation pursuit regression. Acknowledge neural networks, for example. Figure 5b illustrates the type of nonlinear decision boundary that a neural network might find for the loan dataset. In terms of model evaluation, although networks of the convenient amount can universally estimation of any uniform function to any desired degree of accuracy, relatively little is known about the delegation properties of fixed-content networks estimated from the given finite data sets. Also, the usual squared error and cross-entropy undoing functions used for neural networks can be watch as log-likelihood functions for regression and classification respectively Back propagation is a parameter-search method that performs gradient descent in parameter (weight) space to find a local maximum of the likelihood function starting from random initial conditions. Nonlinear regression methods, although tenthly in delegational power, can be hard to interpret.

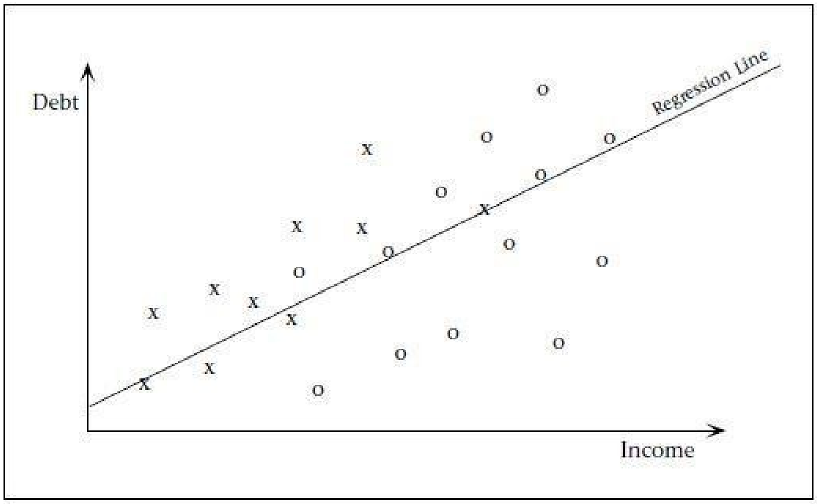


Figure 5a: Simple Linear Regression of Loan data set

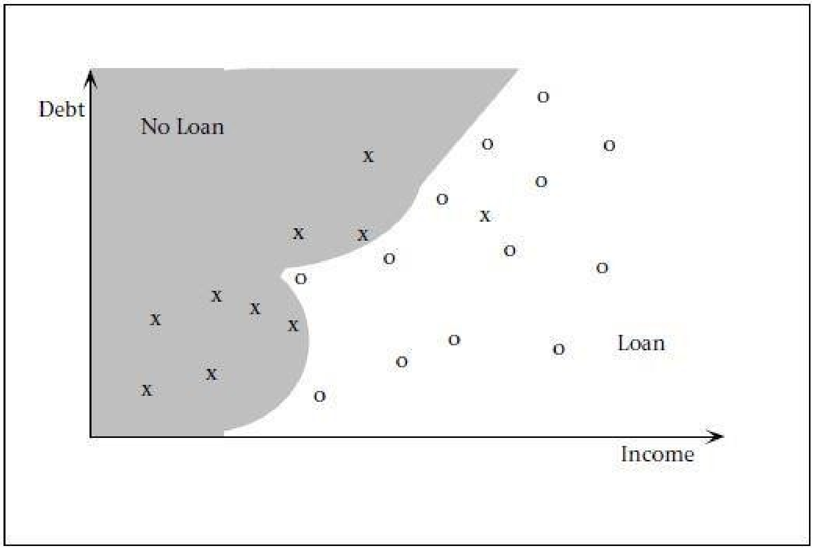


Figure 5b: Simple Linear Regression of Loan data set

2.2.2 Clustering Method:

Clustering is a trivial illuminating task where one expllore to identify a finite set of categories or clusters to describe the data The categories can be commonly exclusive and exhaustive or consist of a richer delegation, such as stratified or overlapping categories.
Summarization involves methods for finding a compact description for a subset of data. A simple example would be tabulating the mean and standard deviations for all fields. More sophisticated methods involve the derivation of summary rules, multivariate visualization techniques, and the discovery of functional relationships between variables. Summarization techniques are often applied to interactive exploratory data analysis and automated report generation. Dependency modeling consists of finding a model that describes significant dependencies between variables. Dependency models exist at two levels:

1-the structural level of the model specifies (often in graphic form) which variables are locally dependent on each other

2 -the quantitative level of the model specifies the strengths of the dependencies using some numeric scale.

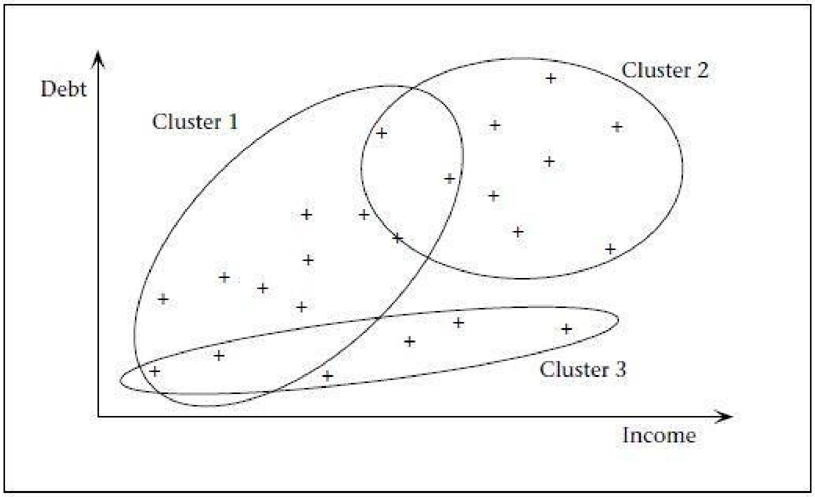


Figure 6: Simple Clustering of Loan data

set 2.2.3 K means Method:

The Algorithm. K-means is one of the elementary unsupervised learning algorithms that solve the well known clustering problem. The strategy follows a smooth and easy way to classify a given data set through a possitive number of clusters (estimating **k** clusters) fixed a priori. The basic idea is to define k centroids, one for each cluster. These centroids should be placed in a tricky way because of various location grounds different result. So, the superior way is to place them as much as conceivable far away from each other. The next step is to pickup each point which belonging to a given data set and relate it to the nearest centroid. When no point is ominous, the first step is finished and an initial grouping is done. At this point we need to re-calculate k new centroids as centers of the clusters that is resulting from the foregoing step. After we have these k new centroids, a new grouping has to be done between the equal data set points and the closest new centroid. A loop has been generated. As a result of this loop we may see that the k centroids relocate their location step by step until no more exchanges are done. In other words centroids do not variaton any more. Finally, this algorithm aims at minimizing an aim function, in this case a squared error as function. The algorithms initially starts with initial estimates for the K centroids, which can either be randomly generated or randomly selected from the data set. The algorithm then emphasize between two steps

(1) Data assignment step

(2) Centroid update step

2.2.3.1 Distance measures in K means

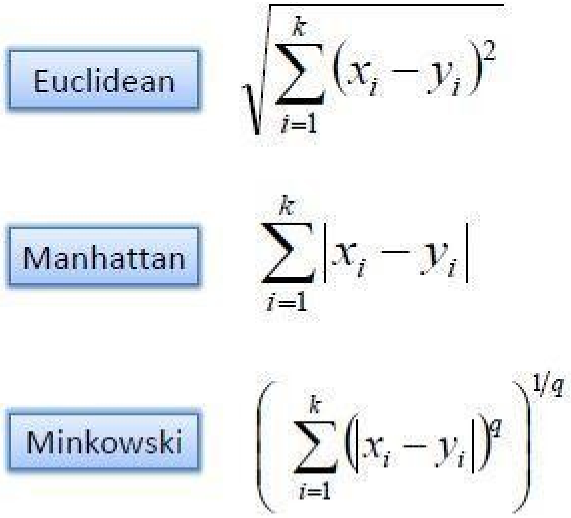

Euclidean
$\sqrt{\sum_{i=1}^{k}(x_i - y_i)^2}$
Manhattan
$\sum_{i=1}^{k}|x_i - y_i|$
Minkowski
$\left(\sum_{i=1}^{k}(|x_i - y_i|)^q\right)^{1/q}$

Figure 7: Comparison between Euclidean, Manhattan and Minkowski measure

Euclidean distance or Euclidean metric is a normal straight-line distance between two points in Euclidean space. With that distance, Euclidean space turn into a measured space. The correlated rules is called the Euclidean rules.

Manhattan Distance is defined as the distance between two points in a grid based on a normal horizontal and/or vertical path, as adverse to the diagonal or "as the strut flies" distance. The Manhattan distance is the normally sum of horizontal and vertical components.

Minkowski distance is a measures in a ruled vector space which can be considered as a deducement of both the Euclidean and Manhattan distances.

2.2.3.2 Advantages and Disadvantages

Advantages:

1. Easily implemented
2. An instance change clusters.
3. With large number of variable k means conputes faster

Disadvantages:

1. Difficult to estimate k value
2. Initial seed have a strong effect in final result

2.2.4 Example Based Method:

The delegation is easily: Use delegation examples from the database to convenient a model that is, estimations on new examples are extracted from the properties of comparable examples in the model whose estimation is known. Techniques include nearest neighbor classification and regression algorithms and case-based rationalizing systems. Figure 8 shows the use of a nearest-neighbor classification for the given loan data set. The class at any new point in the two-dimensional space is the same as the class of the nearest points in the given original training data set. A measure disadvantage of example-based methods (correlated with tree-based methods) is that a well-defined distance measured for calculating the distance between given data points is needed. For the loan data in figure 8, this would not be a problem because income and debt are measured in the same parameters. However, if one wished to include instances such as the duration of the loan, sex, favorites and profession, then it would needed more effort to define a sensible measure between the instances. Model interpretation is basically based on cross-validation calculation of a estimation error. Parameters of the model to be estimated can include the total number of neighbors to use for estimation and the distance measure itself. Like nonlinear regression methods, example-based methods are often asymptotically all-powerful in terms of estimating properties but, conversely, can be difficult in interpretation because the model is implicit data and not explicitly formulated.

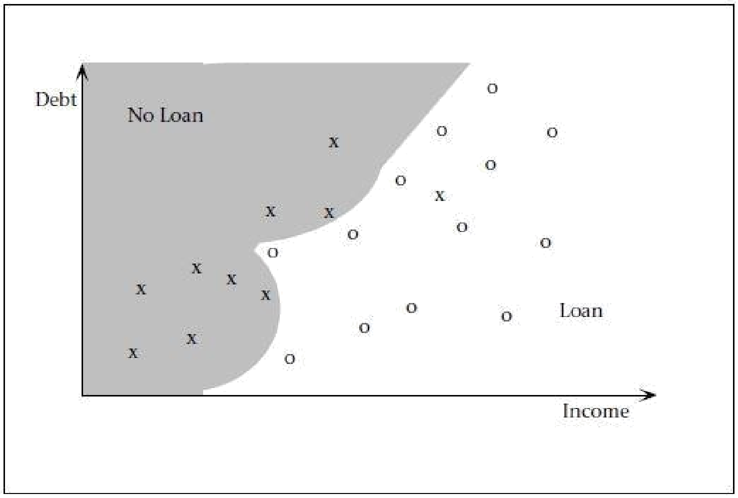


Figure 8: Non Linear method

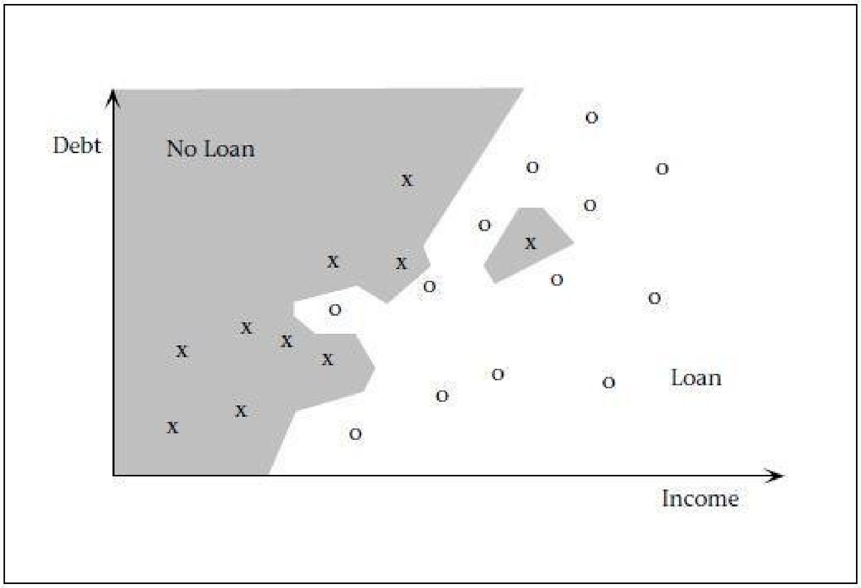


Figure 9: Example based method

Although decision trees and rules have a delegation limited to harmonious logic, relational learning (it is also known as proportional logical programming) uses the more frequent pattern Terminology of first-order logic. A relational learner can easily find formulas such as X = Y. Research to date on model-interpretation methods for analytical learning is logic in nature. The extra delegational potential of analytical models comes at the price of knowing computational exactions in terms of search. Decision trees and rules that use tedious breach have a easy delegational form, making the implied model relatively simple for the user to apprehend.

However, the limitation to a specific tree or norms delegation can significantly limits the functional form (and, thus, the estimation potential) of the model. For example, the effect of a threshold split applied to the income variable for a loan data set: It is clear that using such simple threshold breach (parallel to the feature axes) rigorously limits the type of classification of the outer boundaries that can be included. If one extends the model space to allow more general expressions (such as multifarious hyper planes at inconsistent angles), then the model is more all-powerful for estimation but can be much more difficult to apprehend. A large number of decision tree and rule-induction algorithms are describe in the machine learning and applied statistics literature.

2.2.5 Probabilistic Graphic dependency Model:

Graphic models determine probabilistic dependencies using a graphical structure (Whittaker 1990; Pearl 1988). In its elementary form, the model determine which instanceses are directly dependent on each other. Sometimes, these models are used with categorical or discrete-valued instances, but extensions to special cases, such as Gaussian densities, for real-valued variables are also possible. Within the AI and statistical neighborhood, these models were initially developed within the skeleton of probabilistic expert systems; the structure of the model and the parameters (the conditional probabilities attached to the links of the graph) were elicited from experts. Recently, there has been significant work in both the AI and statistical neighborhood on methods whereby both the structure and the features of graphic models can be learned directly from databases (Buntine 1996; Heckerman 1996). Model-interpretation criteria are basically Bayesian in form, and arguments estimation can be a association of nearest-form estimates and repeated methods depending on whether a variable is directly observed or hidden. Model search can consist of greedy hill-climbing methods over various graph structures. Previous knowledge, such as a partial ordering of the instances based on irregular relations, can be useful in terms of reducing the model search space. Although still mostly in the research phase, graphic model induction methods are of particular interest to KDD because the graphic form of the model lends itself simple to human interpretation. When modeling a biological system, we are interested in entities that are involved in the system under study (e.g., genes) and their different attributes (e.g., expression level).

In a probabilistic model, we treat each of these attributes as a random variable. Random variables include observed attributes, such as the expression level of a particular gene in a particular experiment, as well as hidden attributes that are assumed by the model, such as the cluster assignment of a particular gene. A model embodies the description of the joint probability distribution of all the random variables of interest. Probabilistic graphical models represent multivariate joint probability distributions via a product of terms, each of which involves only a few variables. The structure of the product is represented by a graph that relates variables that appear in a common term. This graph specifies the product form of the distribution and also provides tools for reasoning about the properties entailed by the product. For a sparse graph, the representation is compact and in many cases allows effective inference and learning. In Bayesian Networks, the joint distribution over a set X _ $\{X1, \ldots, Xn\}$ of random variables is represented as a product of conditional probabilities. A Bayesian network associates with each variable Xi a conditional probability *P*(*Xi*_**U***i*), where **U***i* _ *X* is the set of variables that are called the parents of *Xi*. Intuitively, the values of the parents directly influence the choice of value for *Xi*. The resulting product is of the form

P(X1, …, Xn) _ _iP(Xi_**U**i) -------------------- 1

The graphical representation is given by a directed graph where we put edges from Xi's parents to Xi. If the graph is acyclic, the product decomposition of Eq. 1 is guaranteed to be a coherent probability distribution.

Bayesian networks appear naturally in several domains in biology. In analysis, for example, the joint distribution of genotypes in a pedigree is a product of conditional probabilities of the genotype of each individual given the genotypes of its two biological parents. In phylogenetic models, the probability over all possible sequences during evolution is the product of the conditional probability of each sequence given its latest ancestral sequence in the phylogeny.

To specify a model completely, we need to describe the conditional probability associated with each variable. In general, any statistical regression model may be used.

For example, we can consider models where each *P*(Xi_**U**i) is a linear regression of Xi on **U**i. Alternatively, we can use decision trees to represent the probability of a discrete variable *X*i given the values of its parents. The choice of a specific parametric representation of the conditional probabilities is often dictated by our knowledge or assumptions about the domain.

2.3 Representation

2.3.1 Model Representation:

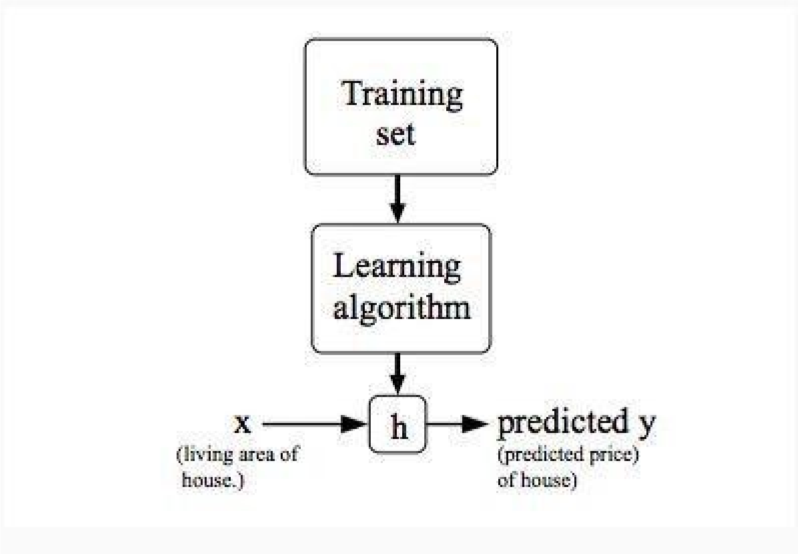


Figure 10: Model Representation

2.3.2 Model Interpretation:

Model Interpretation is an essential part of the model development process. It benefits to find the best model that delegate our information and how well the selected model will work in the future. Interpreting model is the realization with the information used for training is not acceptable in data mining because it can easily generate over optimistic and over fit models.

2.3.3 Search:

Search analysis takes place by automatically checking huge amount of data generating from internet usage statistics, keyword usage statistics and many other parameter and reasons. Data mining tools can keep record of this data by frequently storing them, analysis them and generating outcomes as and when needed. A number of websites provide frequent to use tools for analyzing your piece and contentment so that the search engines can estimate them on their starts pages, but hardly do we know about that they use data mining applications on a smaller scale. There are numerous questions that rise up in our mind such as, how these tools mine the data, where are these information stored, how is the transform done, and more. The most important one is to provide real-time analysis that makes it obvious to use data mining

Chapter-3 System Development

3.1 KNN Introduction:

Definition:

1. K nearest neighbors is a smooth algorithm that stores all available cases and classifies new cases based on a parallel measure.
2. KNN has been used in statistical prediction and pattern recognition.
3. A non-parametric lazy technique.

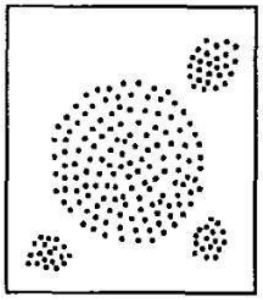
3.2 Assumption in KNN

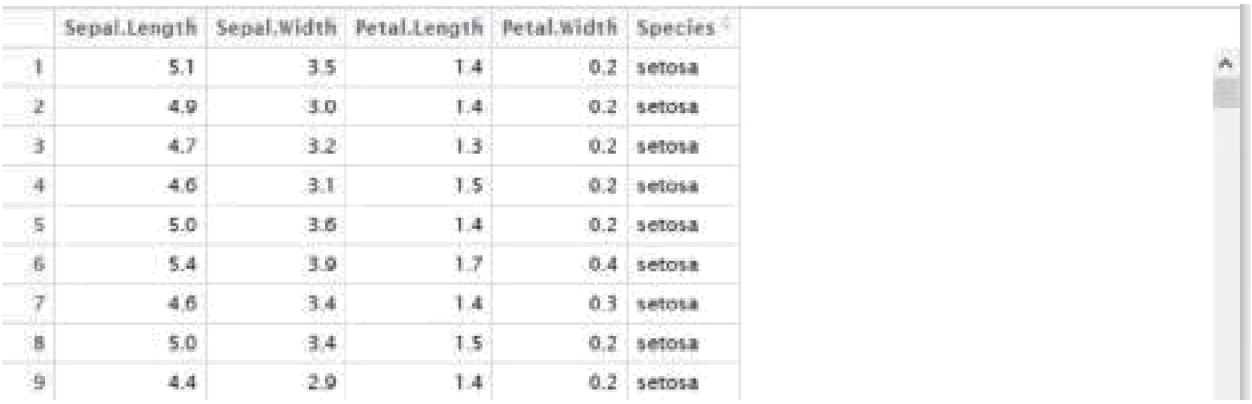
1. All the instances correspond to point in an n-dimensional space.
2. Each set is represented with a set of numerical attributes.
3. Each of the training data consist of a set of vectors and class labels associated with each vector.
4. Classification is done by comparing features vectors of different K nearest points.

3.3 KNN for Density Estimation:

DBSCAN (Density-Based dimensional clustering and application), is a density-based clustering algorithm classification, made known in Ester et al. 1996, which can be used to specify clusters of any outline in a data set consist of noise and outliers.

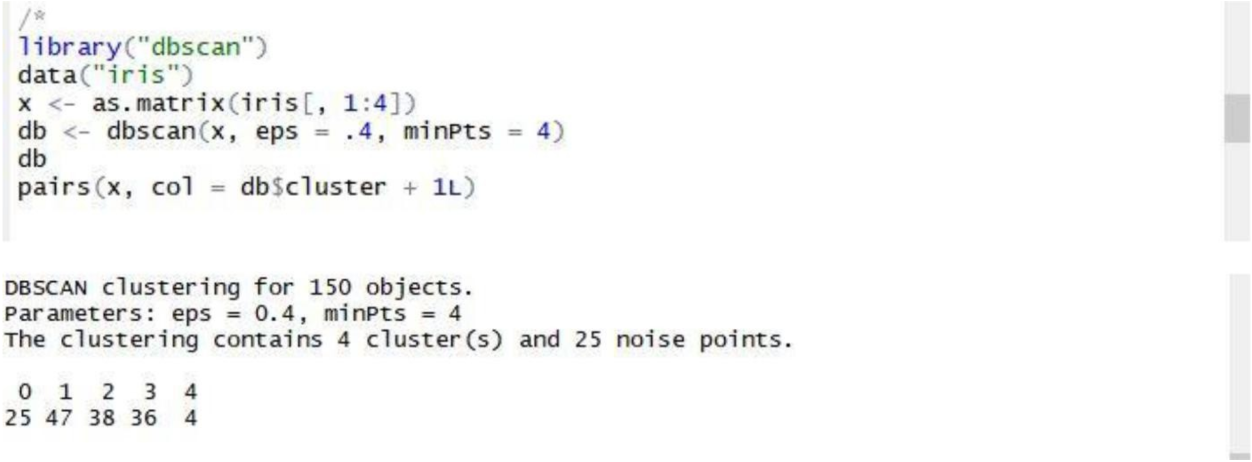
The key idea trailing the density-based clustering approach is imitative from a human innate clustering method. Clusters are heavily dense regions in the data space, are distinct by regions of lower density of points. The DBSCAN algorithm is basically on this innate assumption of clusters and noise. The key idea is that for each point of a given cluster, the closed of a given radius has to consist of at least a minimum number of points. Clusters are heavily dense regions in the data space, separated by regions of lower density of points.





	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5.0	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa

Table 1: Dataset



```
/*
library("dbscan")
data("iris")
x <- as.matrix(iris[, 1:4])
db <- dbscan(x, eps = .4, minPts = 4)
db
pairs(x, col = db$cluster + 1L)
```

```
DBSCAN clustering for 150 objects.
Parameters: eps = 0.4, minPts = 4
The clustering contains 4 cluster(s) and 25 noise points.

 0  1  2  3  4
25 47 38 36  4
```

Figure 11: Example Density method

3.3.1 Advantages and Disadvantages

Advantages:

1. Not to define number of clusters
2. Find clusters on arbitary basis
3. Clusters surrounded by a different cluster

Disadvantages:

1. Depend on distance measure
2. Cannot cluster data on basis of density

3.4 KNN Classification:

KNN is a algorithm that stores all convenient cases and specify new cases based on a parallel measure (e.g., distance functions, density functions). K nearest neighbor has been used in prediction and recognizing interesting pattern previously in the beginning of 1970's as a non-parametric technique. In this case, we are given some data points for training and also a new unleveled data for testing. Our basic aim is to find the class label for the new point. The algorithm has different behavior based on k.

3.4.1 Case 1: K=1:

This is the simple scenario. Let x be the point to be labeled . Find the point nearest to x . Let it be y. Now nearest neighbor rule asks to assign the label of y to x. This seems too easy and some times even counter innate. If you feel that this process will result a enormous error, you may be right – but there is a catch for solution. This interpretation holds only when the number of data points is not enormous. If the number of data points is numerous, then there is a numerous chance that label of x and y are similar. An example – Lets say you have a biased coin. You toss it for 1 million time and you have got head 10,00,000 times. Then most expected your next call will be head. We can use a almost same parameters here. Let me try an informal parameter here - Assuming all points are in a D plane . The number of points is acceptability is enormous. This means that the density of the plane at any point is fairly enormous. In other words, within any subspace there is competent number of points. Consider a point x in the space which also has a number of neighbors. Now let y be the nearest neighbor. If x and y are very close, then we may assume that probability that x and y belong to similar class is fairly similar – Then by decision theory, x and y have the same class.

3.4.2 Case 2: k=K:

This is a forward extension of 1NN. Simply what to do is that we try to find the k nearest neighbor and do a enormous voting. Consistently K is odd when the number of classes is 2. Let say k = 5 and there are 2 instances of n1 and 3 instances of n2. In this case, K nearest neighbor says that new point has to labeled as n2 as it forms the enormous closeness. We follow a same parameter when there are numerous classes. One of the forward expansion is not to give one vote to all the neighbors. A very common thing that we do is weighted KNN where each point has a weight which is consistently recalculated using the given distance. For Example under inverted distance weighting, each point has a weight equal to the inverted of its distance to the point to be classified. This means that neighboring points have a enormous vote than the farther points. It is quite visible that the accuracy may or may not increase when we increase k but the computing cost also increases.

3.5 KNN Working (how KNN algorithm works):

1. A point is classify by enormous votes of its nearest neighbor classes.
2. The point is assigned to most similar class among its k nearest neighbors.
3. K Nearest Neighbor algorithm calculates the distance between the query point X and a set of training samples Y to classify the object in majority of k nearest neighbors.

- Query Point $X_i=X_1,X_2,X_3……….,X_n$

  Training Sample $Y_i=Y_1,Y_2,Y_3………….,Y_n$

3.5.1 Advantages and Disadvantages

Advantages:

1. Very simple and intuitive.
2. Can be applied to the data from any distribution.
3. Good classification if the number of samples is large

Disadvantages:

1. Take more time to classify a new sample.
2. Need to calculate and compares distance from to all others examples.
3. Choosing K may be tricky.
4. Neighbor elements are too away from test element

3.6 K value (How to choose):

1. If K is too small it is case sensitive to noise points.
2. Large K works well, but too large K may include majority of points from others of classes.

3. Rule of thumb K< sqrt (n),n is number of examples

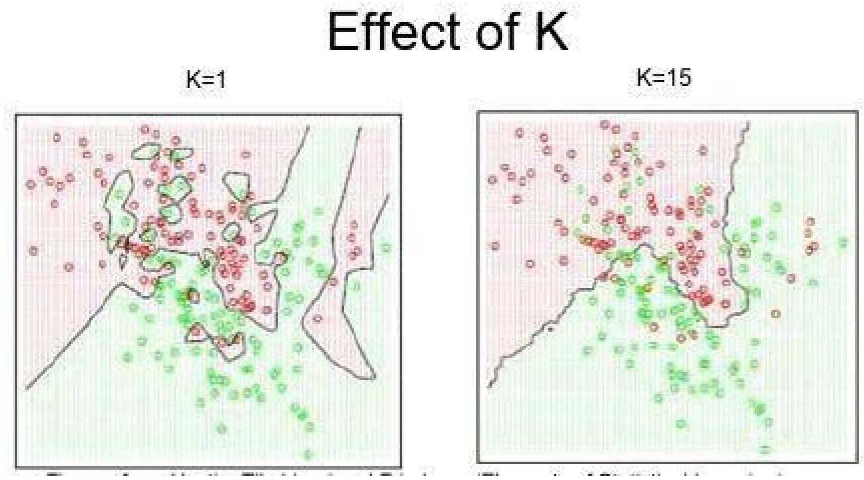
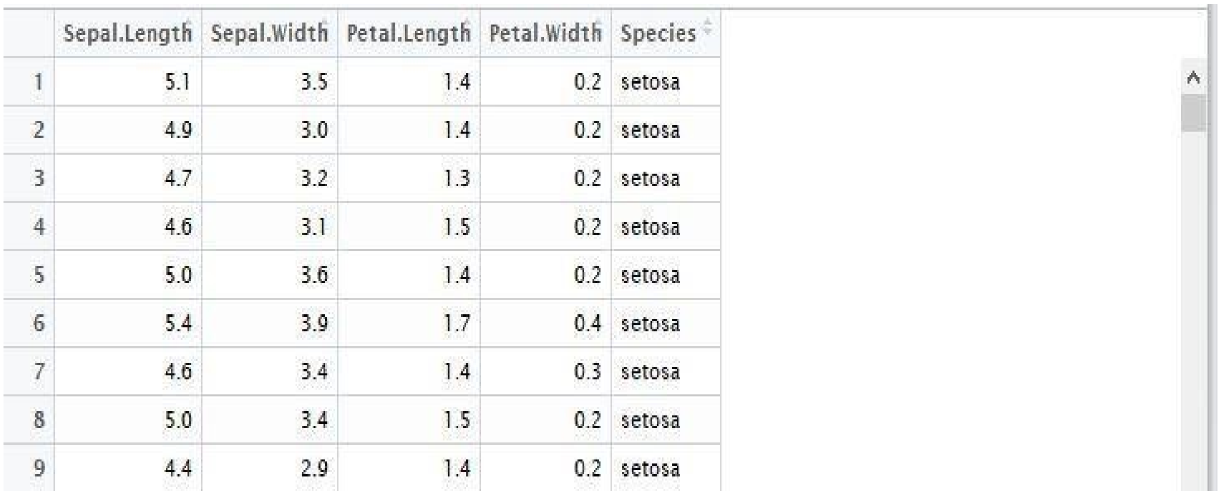


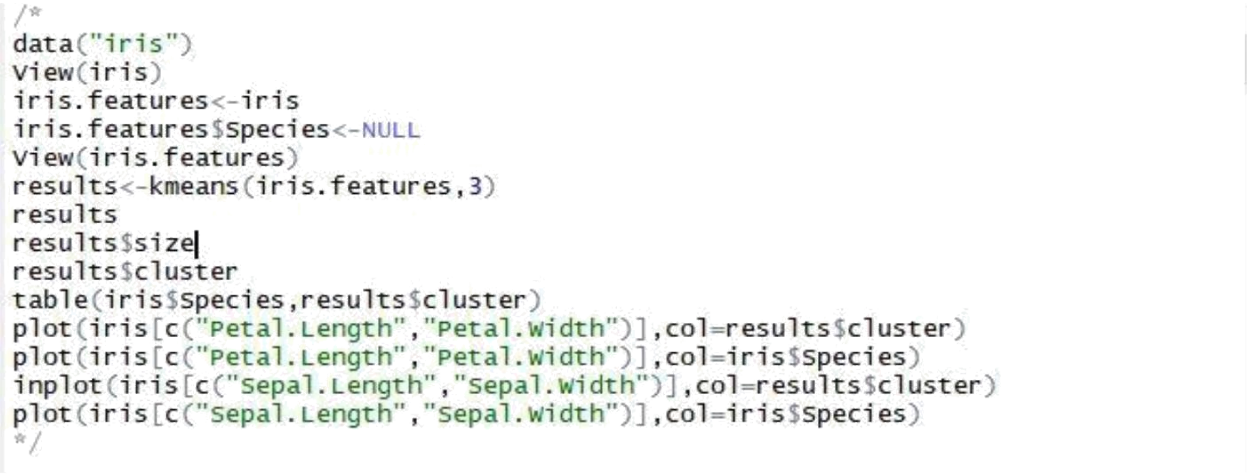
Figure 12: Clustering on k value

3.7 K means Method:



	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5.0	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa

Table 2: Dataset



```
/*
data("iris")
View(iris)
iris.features<-iris
iris.features$Species<-NULL
View(iris.features)
results<-kmeans(iris.features,3)
results
results$size
results$cluster
table(iris$Species,results$cluster)
plot(iris[c("Petal.Length","Petal.Width")],col=results$cluster)
plot(iris[c("Petal.Length","Petal.Width")],col=iris$Species)
inplot(iris[c("Sepal.Length","Sepal.Width")],col=results$cluster)
plot(iris[c("Sepal.Length","Sepal.Width")],col=iris$Species)
*/
```

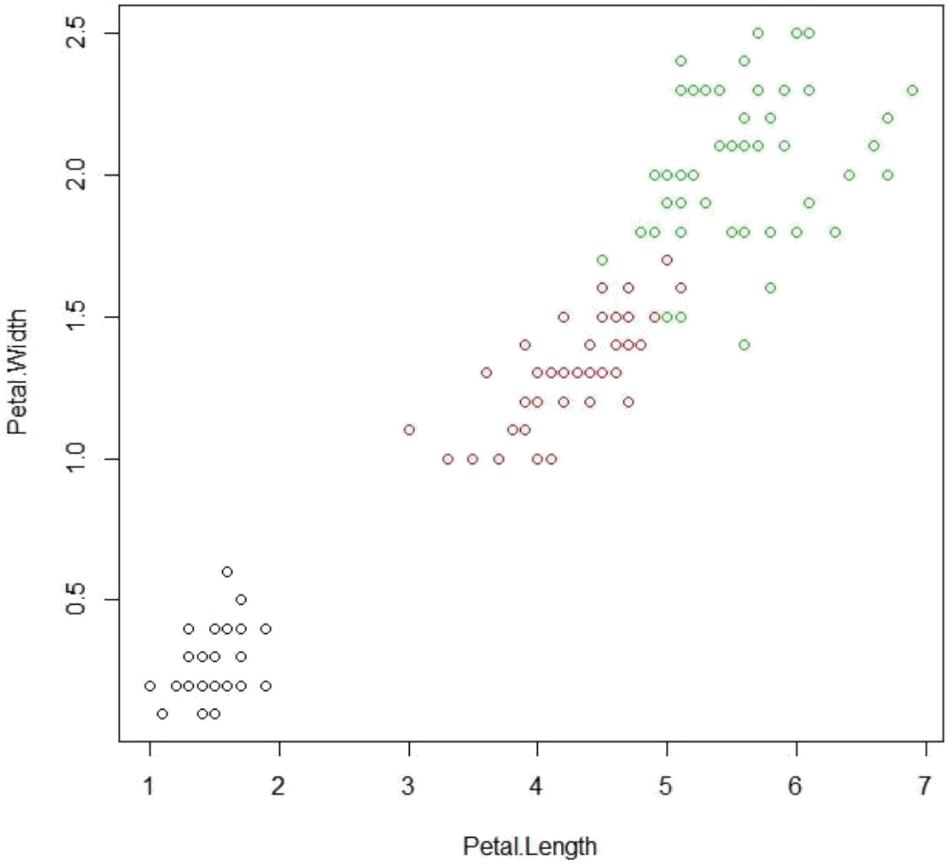


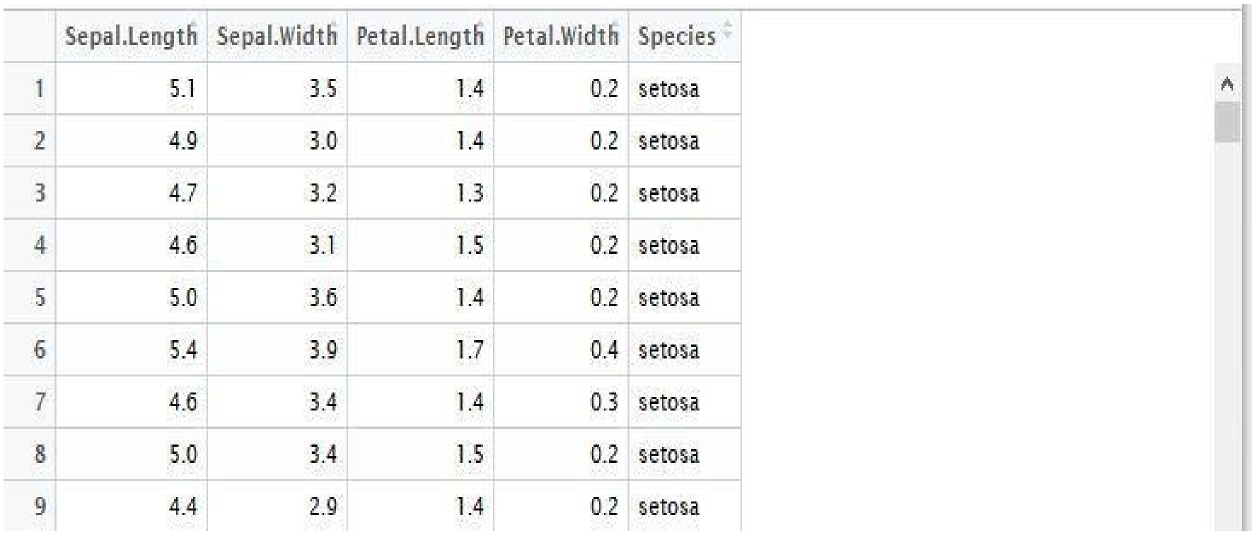
Figure 13: Example K means method

3.8 DensityClust Method:

DensityClust method the user with machanism for generating the initially rho and delta values for each result as well as using these to assigning calculated results to clusters. This can be done in two steps the user is freebie for assigning of results to clusters using a new set with rho and delta thresholds values, No need to recalculate everything.

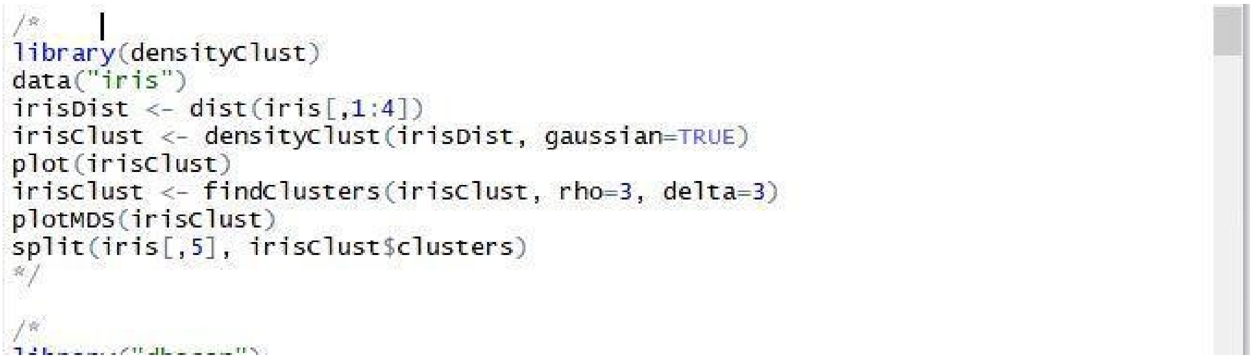
There are Two plots are sustained by the package, and both the plots used in the declaration for the algorithm. The specific plot function generate a output plot, with different or various shades of peaks of clusters, if these are assigned. plotMDS() performs a intricate scaling of the distance matrix and plots as a scatter plot. If clusters are assigned results and shaded according to their values.

There are two functions in the package these are DensityClust() and FindClusters().In first step, pickup a distance matrix and a cutoff and calculates rho and delta for each result. The later takes the output of densityClust() which make cluster assignment for each result obtain after calculation based on a user defined values.



	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5.0	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa

Table 3: Dataset



```
/*
library(densityClust)
data("iris")
irisDist <- dist(iris[,1:4])
irisClust <- densityClust(irisDist, gaussian=TRUE)
plot(irisClust)
irisClust <- findClusters(irisClust, rho=3, delta=3)
plotMDS(irisClust)
split(iris[,5], irisClust$clusters)
*/

/*
library("dbscan")
```

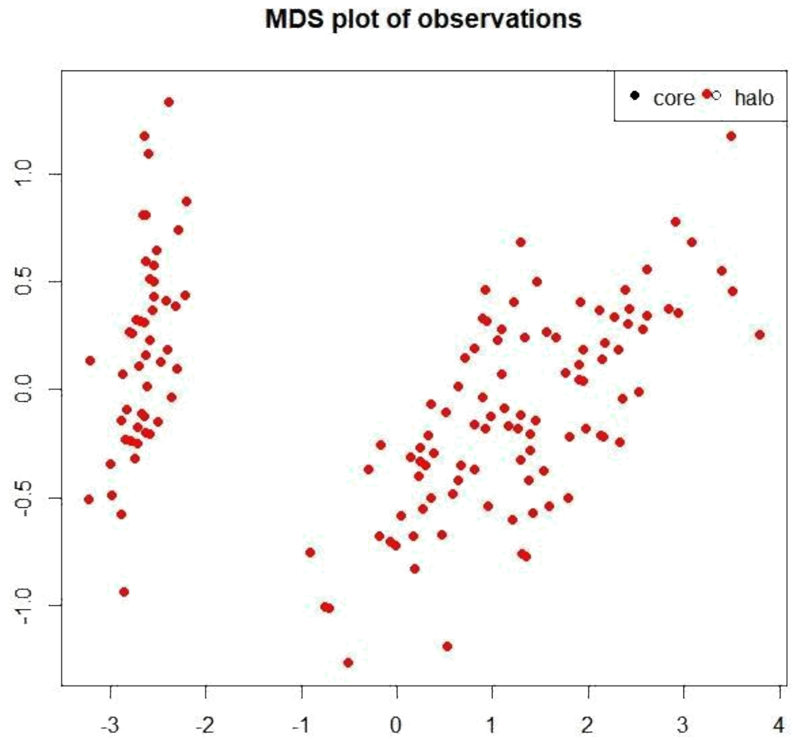


Figure 14: Example DensityClust method

3.8.1 Advantages

Advantages:

1. More acurate in finding clusters in respect of DBSCAN
2. Overcome Euclidean Distance
3. Find clusters on local density

3.9 WKNN Method:

Weighted K-Nearest Neighbor Classifier: Performs KNN classification of a test set with the use of training set. For each entry of the test set, the k nearest neighbor training set vectors (rendering to Minkowski distance) are calculated, and the classification is done on the basis maximum of sum of kernel densities. In addition even primitive and continuous instances are estimated.

1- kknn(formula = formula(train), tn, test, na.action = na.omit(), k = 7, ykernel = NULL distance = 2, kernel = "optimal" , contrasts = c('unordered' = "contr.dummy", ordered = "contr.ordinal"), scale=TRUE,)

2-kknn.dist(learn, valid, k = 10, distance = 2)

Arguments:

Formula: A formula object.

Train: Matrix of training set .

Test: Matrix of test set.

Learn: Matrix of training set

Valid: Matrix of test set

na.action: for data which consist of 'NA's values.

k: Number of neighbor

Distance: Parameter of Minkowski distance.

Kernel: Kernel to use. Possible choices are "rectangular", "triangular", ("epanechnikov" "biweight", "triweight", "cos", "inv", "gaussian", "rank" and "optimal").

Ykernel: length of y-kernel

Scale: scale variable.

Contrasts:'unordered' and 'ordered' contrasts vector to use.

KKNN: Returns a list-object

3.9.1 Advantages

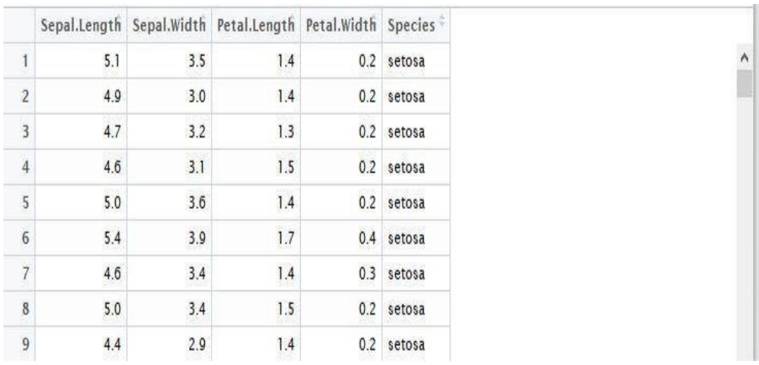
Advantages:

1. Effect is less of interclass standard variation
2. Not too many failures
3. Not affected from neighbour boundary

Disadvantages:

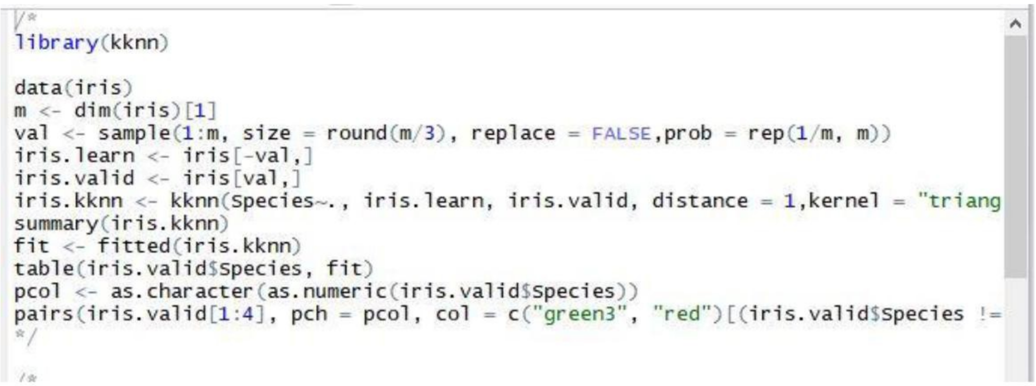
1. Take more time to classify a new sample.

2.. Choosing K may be tricky.



	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5.0	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa

Table 4: Dataset



```
/*
library(kknn)

data(iris)
m <- dim(iris)[1]
val <- sample(1:m, size = round(m/3), replace = FALSE,prob = rep(1/m, m))
iris.learn <- iris[-val,]
iris.valid <- iris[val,]
iris.kknn <- kknn(Species~., iris.learn, iris.valid, distance = 1,kernel = "triang
summary(iris.kknn)
fit <- fitted(iris.kknn)
table(iris.valid$Species, fit)
pcol <- as.character(as.numeric(iris.valid$Species))
pairs(iris.valid[1:4], pch = pcol, col = c("green3", "red")[(iris.valid$Species !=
*/

/*
```

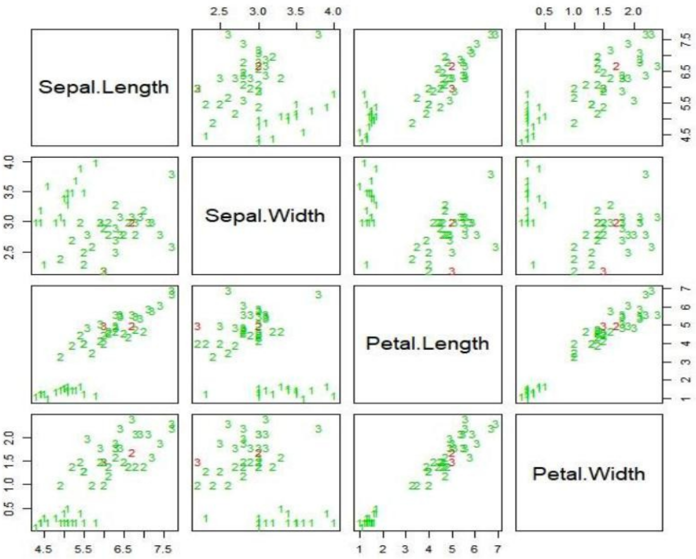
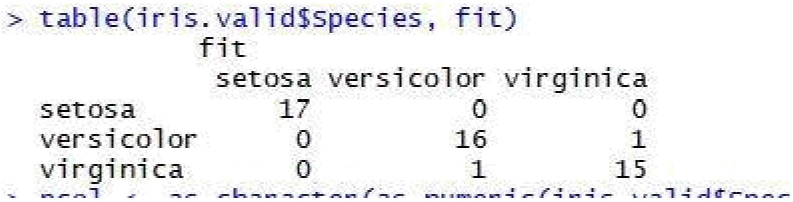


Figure 15: Example WKNN method



```
> table(iris.valid$Species, fit)
            fit
             setosa versicolor virginica
  setosa         17          0         0
  versicolor      0         16         1
  virginica       0          1        15
> pcol <- as.character(as.numeric(iris.valid$Spec
```

Table 5: Example KKNN method

Chapter 4 Performance

4.1 Resource used:

R:

R is a language and an environment for calculation and statistical graphics. It is a project same as the language and environment S, which was developed by Bell Laboratories (formerly AT & T, now Lucent Technologies) by John Chambers and his colleagues. R can be expressed as different way of implementation of S. There are some important differences, but much of the code written for S are not changeable under R.

R provides a large variety of numerical models (linear and nonlinear modeling, classical numerical tests, time series analysis, classification, grouping) and graphs and is enormously extensible. The S language is generally the vehicle of choice for research in numerical techniques and R availables an open way for participation in various activity.

One of R's strengths is the easy with which well-designed publication graphs with high quality can be originated, which involves mathematical symbols and formulas when are needed. Careful attention has been paid to the default values for minor design options in the graphics, but the user maintains complete control.

R is an integrated set of software involves data manipulation, calculation and displaying graphics. It includes the following:

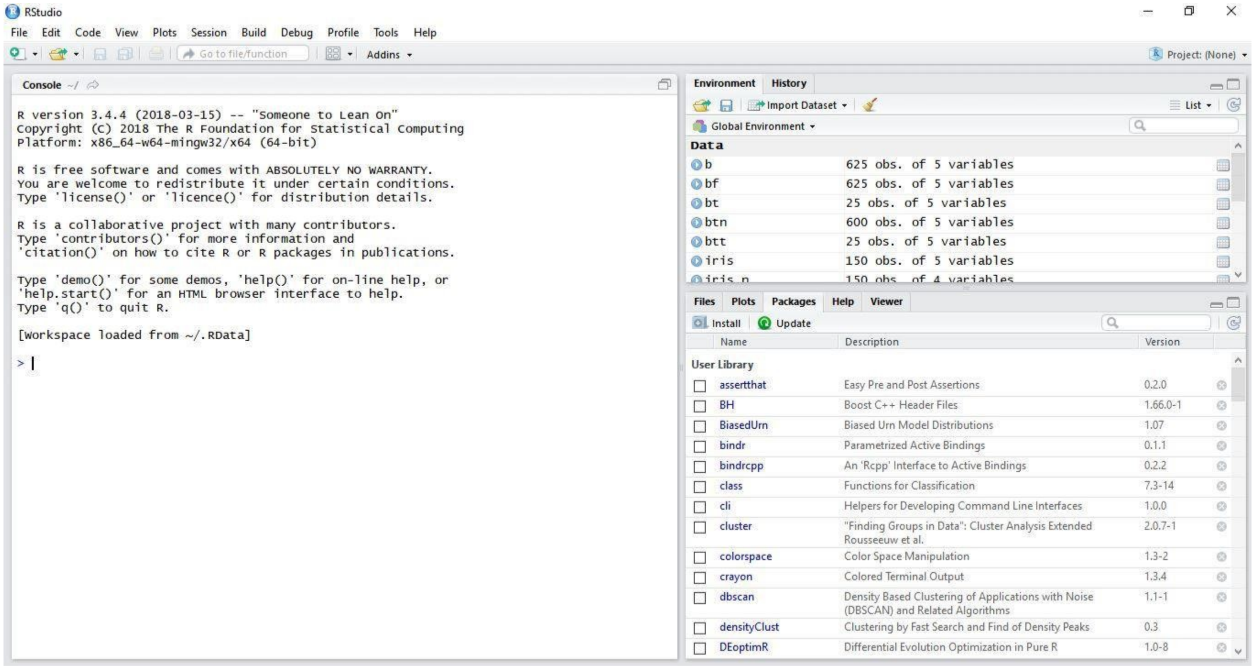
- a system for storing and managing data,
- a set of operators for calculations
- a integrated collection of tools for data analysis,
- graphical structures for data analysis and visualization on screen or on paper

RSTUDIO:

RStudio:

- It is an integrated programming software
- it's an open source software

RStudio



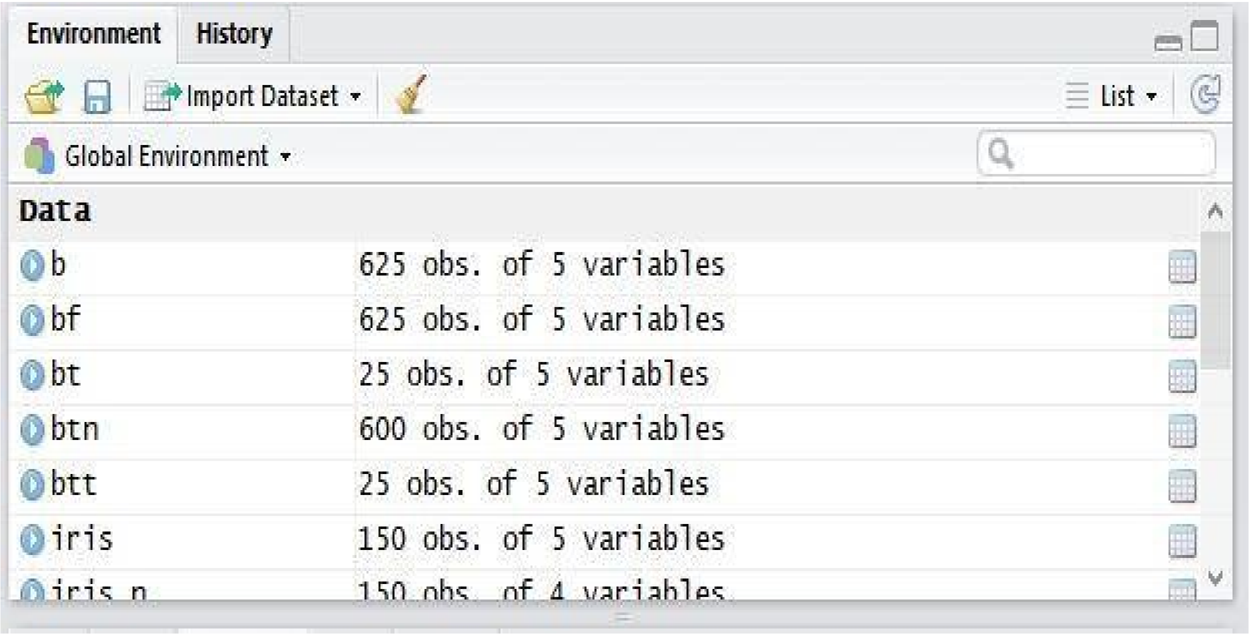
```
R version 3.4.4 (2018-03-15) -- "Someone to Lean On"
Copyright (C) 2018 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

[Workspace loaded from ~/.RData]

> |
```

Data	
b	625 obs. of 5 variables
bf	625 obs. of 5 variables
bt	25 obs. of 5 variables
btn	600 obs. of 5 variables
btt	25 obs. of 5 variables
iris	150 obs. of 5 variables
iris_n	150 obs. of 4 variables

	Name	Description	Version
User Library			
□	assertthat	Easy Pre and Post Assertions	0.2.0
□	BH	Boost C++ Header Files	1.66.0-1
□	BiasedUrn	Biased Urn Model Distributions	1.07
□	bindr	Parametrized Active Bindings	0.1.1
□	bindrcpp	An 'Rcpp' Interface to Active Bindings	0.2.2
□	class	Functions for Classification	7.3-14
□	cli	Helpers for Developing Command Line Interfaces	1.0.0
□	cluster	"Finding Groups in Data": Cluster Analysis Extended Rousseeuw et al.	2.0.7-1
□	colorspace	Color Space Manipulation	1.3-2
□	crayon	Colored Terminal Output	1.3.4
□	dbscan	Density Based Clustering of Applications with Noise (DBSCAN) and Related Algorithms	1.1-1
□	densityClust	Clustering by Fast Search and Find of Density Peaks	0.3
□	DEoptimR	Differential Evolution Optimization in Pure R	1.0-8

Figure 16: Rstudio

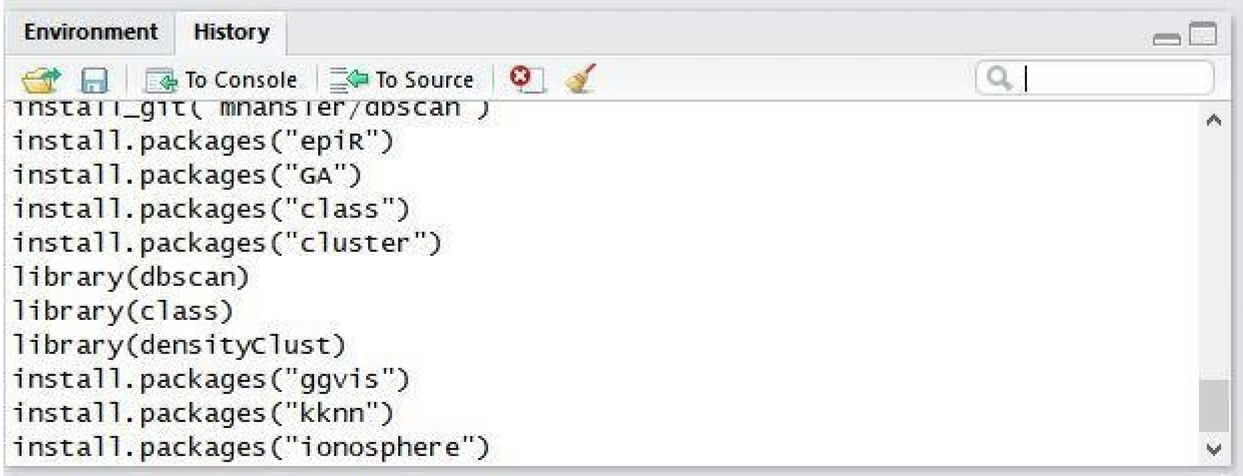
Enviroment: list of R objects



Data	
b	625 obs. of 5 variables
bf	625 obs. of 5 variables
bt	25 obs. of 5 variables
btn	600 obs. of 5 variables
btt	25 obs. of 5 variables
iris	150 obs. of 5 variables
iris_n	150 obs. of 4 variables

Figure 17: Enviroment tab

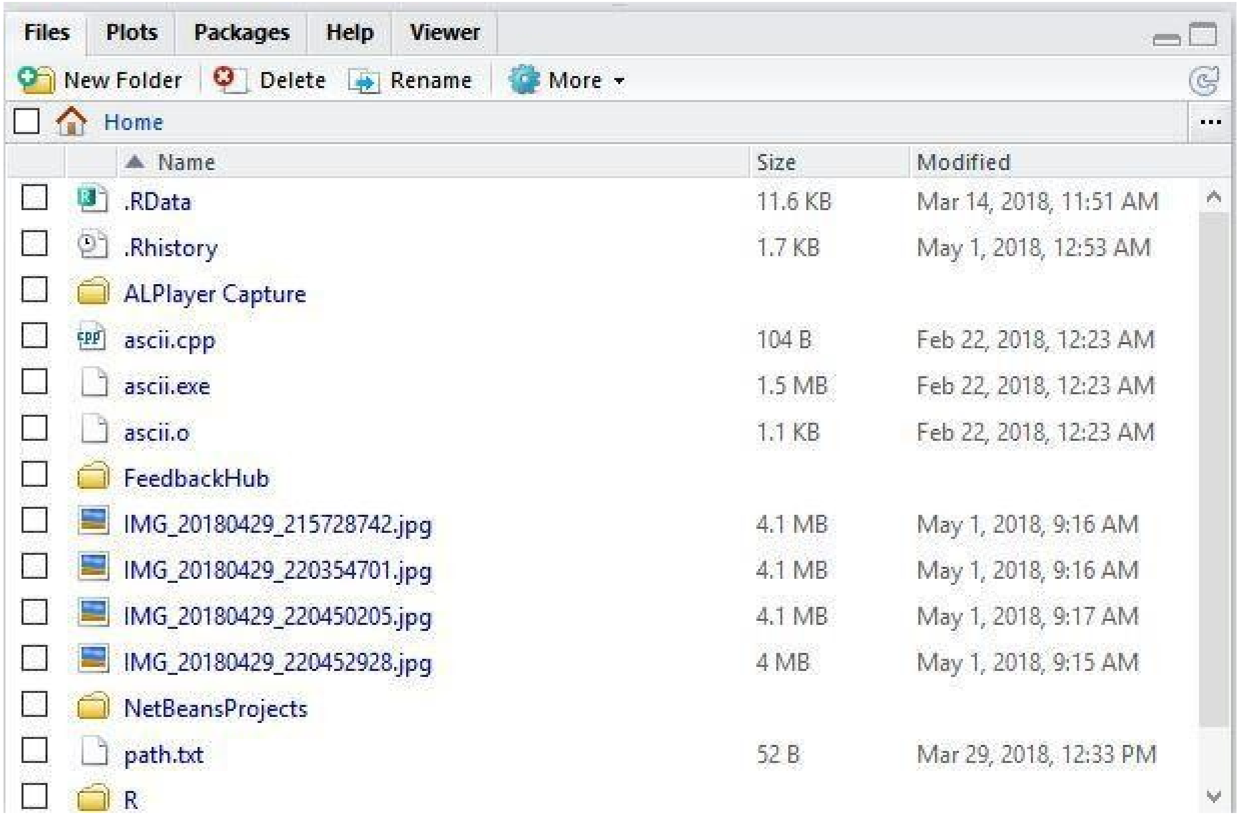
History: list of commands entered into the Console



```
install_git("mhahsler/dbscan")
install.packages("epiR")
install.packages("GA")
install.packages("class")
install.packages("cluster")
library(dbscan)
library(class)
library(densityClust)
install.packages("ggvis")
install.packages("kknn")
install.packages("ionosphere")
```

Figure 18: History tab

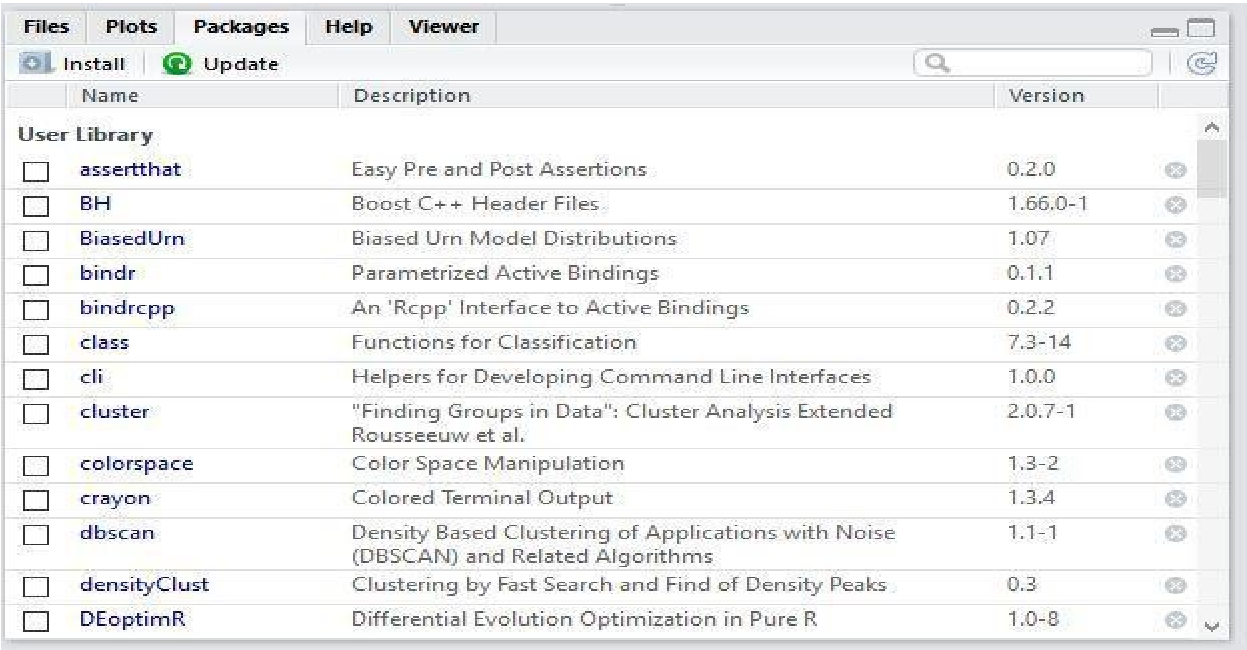
File: where the file store



	Name	Size	Modified
□	.RData	11.6 KB	Mar 14, 2018, 11:51 AM
□	.Rhistory	1.7 KB	May 1, 2018, 12:53 AM
□	ALPlayer Capture		
□	ascii.cpp	104 B	Feb 22, 2018, 12:23 AM
□	ascii.exe	1.5 MB	Feb 22, 2018, 12:23 AM
□	ascii.o	1.1 KB	Feb 22, 2018, 12:23 AM
□	FeedbackHub		
□	IMG_20180429_215728742.jpg	4.1 MB	May 1, 2018, 9:16 AM
□	IMG_20180429_220354701.jpg	4.1 MB	May 1, 2018, 9:16 AM
□	IMG_20180429_220450205.jpg	4.1 MB	May 1, 2018, 9:17 AM
□	IMG_20180429_220452928.jpg	4 MB	May 1, 2018, 9:15 AM
□	NetBeansProjects		
□	path.txt	52 B	Mar 29, 2018, 12:33 PM
□	R		

Figure 19: File tab

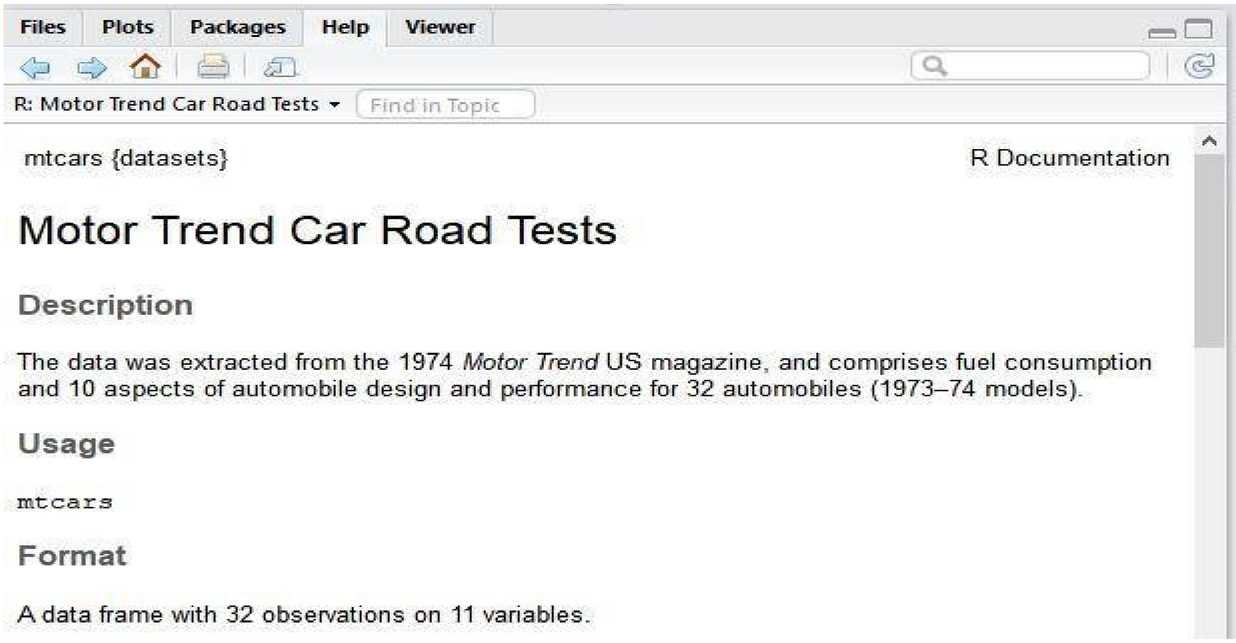
Package: List of installed packages



	Name	Description	Version
	User Library		
□	assertthat	Easy Pre and Post Assertions	0.2.0
□	BH	Boost C++ Header Files	1.66.0-1
□	BiasedUrn	Biased Urn Model Distributions	1.07
□	bindr	Parametrized Active Bindings	0.1.1
□	bindrcpp	An 'Rcpp' Interface to Active Bindings	0.2.2
□	class	Functions for Classification	7.3-14
□	cli	Helpers for Developing Command Line Interfaces	1.0.0
□	cluster	"Finding Groups in Data": Cluster Analysis Extended Rousseeuw et al.	2.0.7-1
□	colorspace	Color Space Manipulation	1.3-2
□	crayon	Colored Terminal Output	1.3.4
□	dbscan	Density Based Clustering of Applications with Noise (DBSCAN) and Related Algorithms	1.1-1
□	densityClust	Clustering by Fast Search and Find of Density Peaks	0.3
□	DEoptimR	Differential Evolution Optimization in Pure R	1.0-8

Figure 20: Packages tab

Help:



mtcars {datasets} R Documentation

Motor Trend Car Road Tests

Description

The data was extracted from the 1974 *Motor Trend* US magazine, and comprises fuel consumption and 10 aspects of automobile design and performance for 32 automobiles (1973–74 models).

Usage

```
mtcars
```

Format

A data frame with 32 observations on 11 variables.

Figure 21: Help tab

4.2 Methods

4.2.1 Different method:

K-Means:

k-means clustering is a approach of vector quantize, primitively from signal processing, that is prevailing for cluster analysis in data mining.

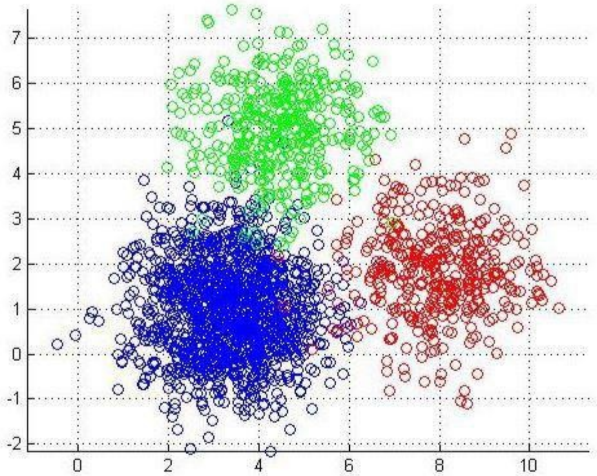
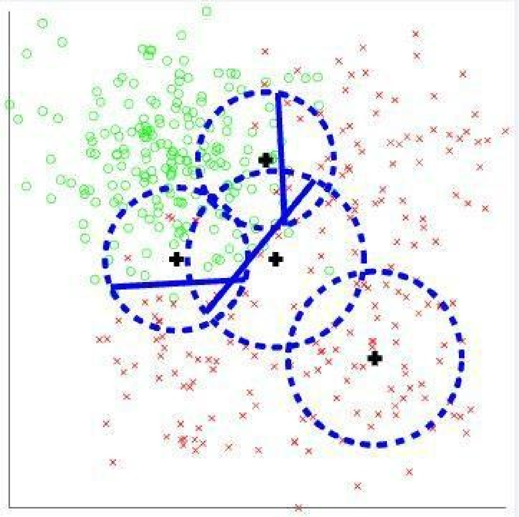


Figure 22: Kmeans

KNN: k-nearest neighbors algorithm (k-NN) is a non-parametric approach used for classification and regression.



.

Figure 23: KNN

DensityClust: An improved implementation (based on k-nearest neighbors) of the density peak clustering algorithm

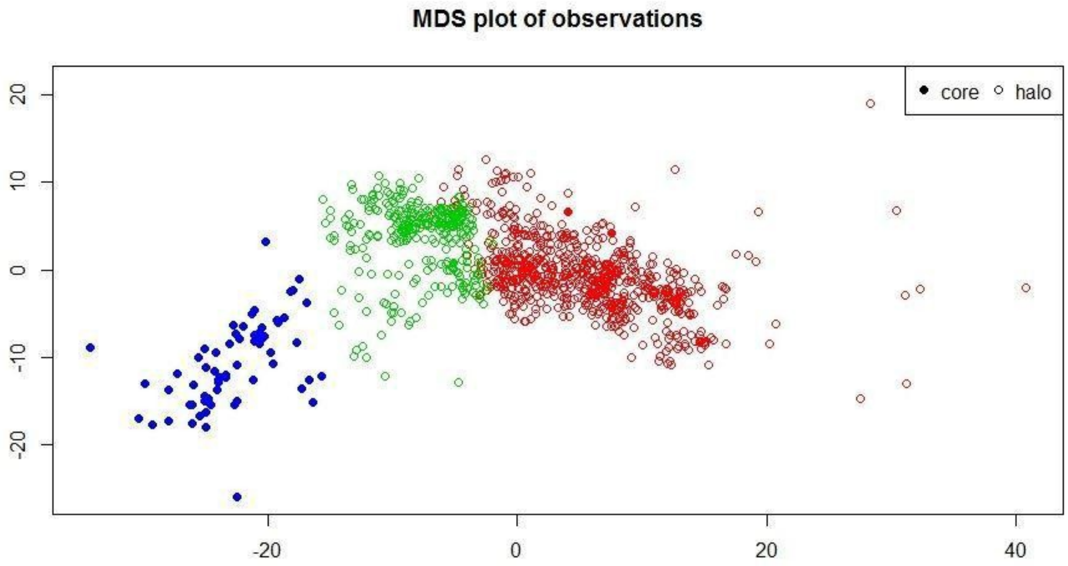


Figure 24: DensityClust

WKNN: Performs KNN classification of test set. For every data of test set, the KNN training set (according to Minkowski distance) are find, and the classification is done on the basis of max of sum of kernel densities. In addition even statistic and connected instances can be estimated.

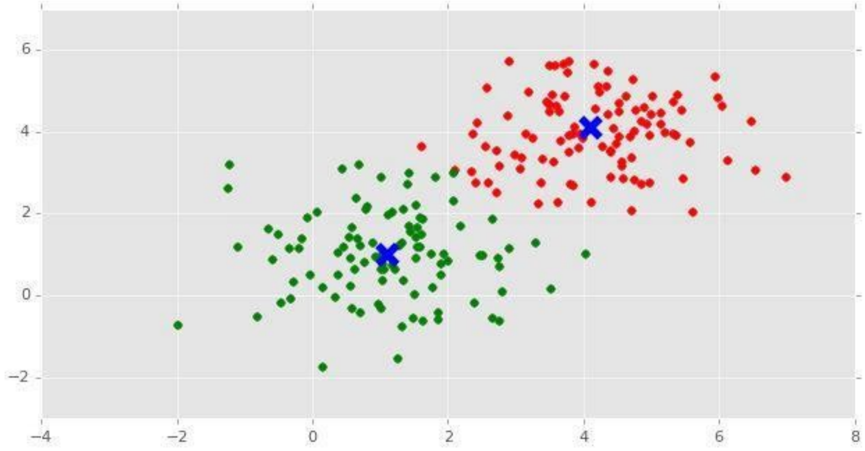
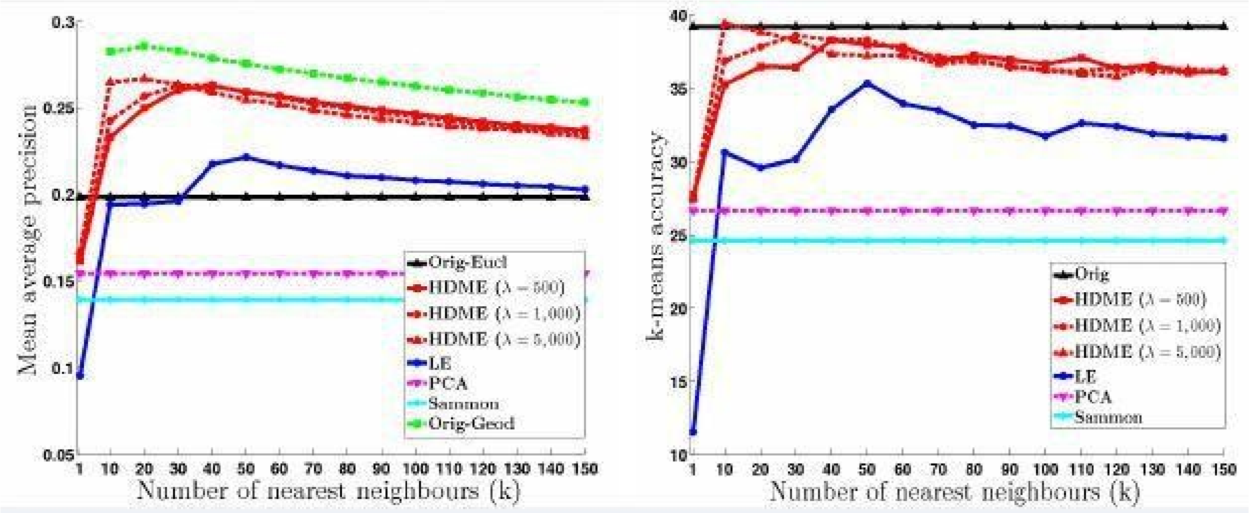


Figure 25: KKNN

4.2.2 Accuracy and complexity

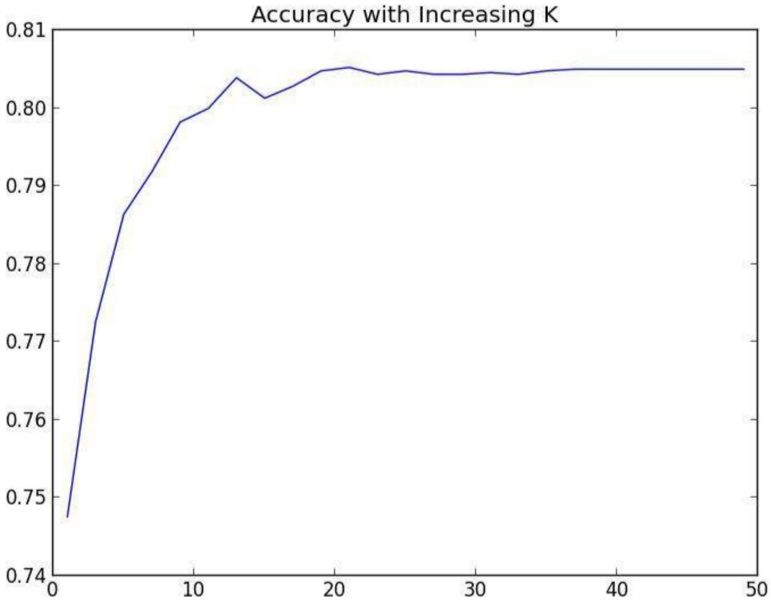
K-Means:



Garph 1: Kmeans

-O(n^2)

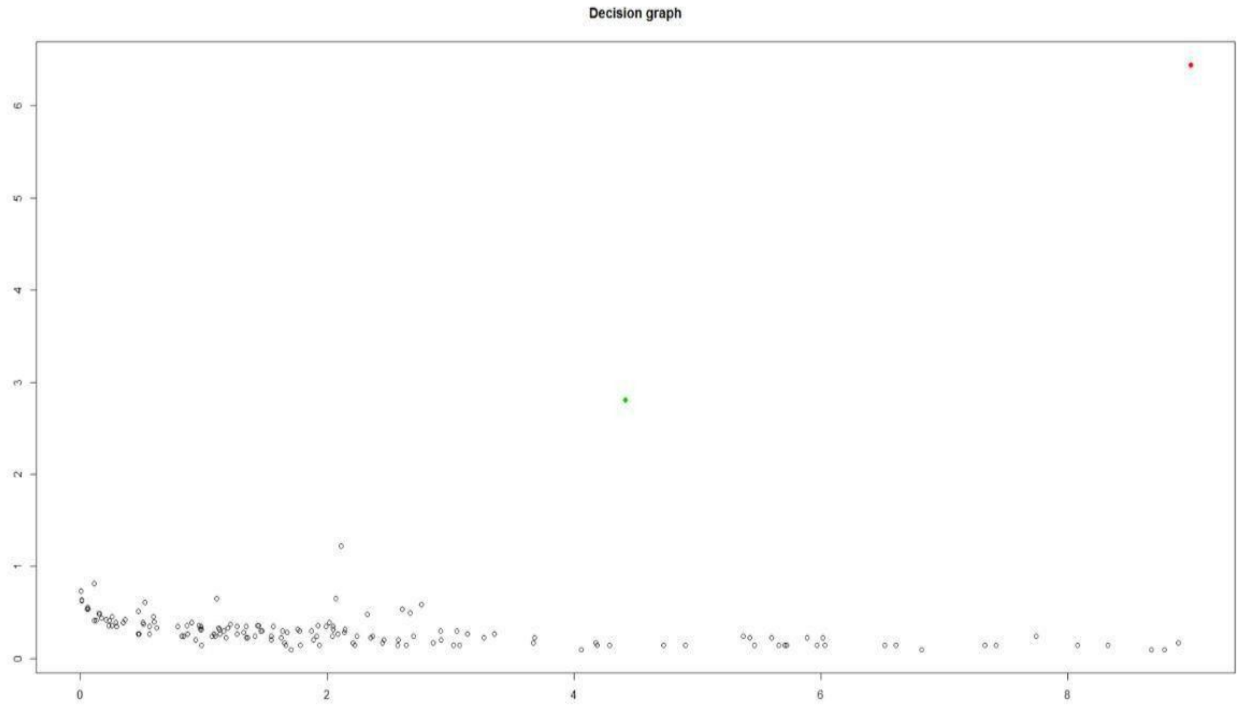
KNN:



Garph 2: KNN

-O(d)

DensityClust:

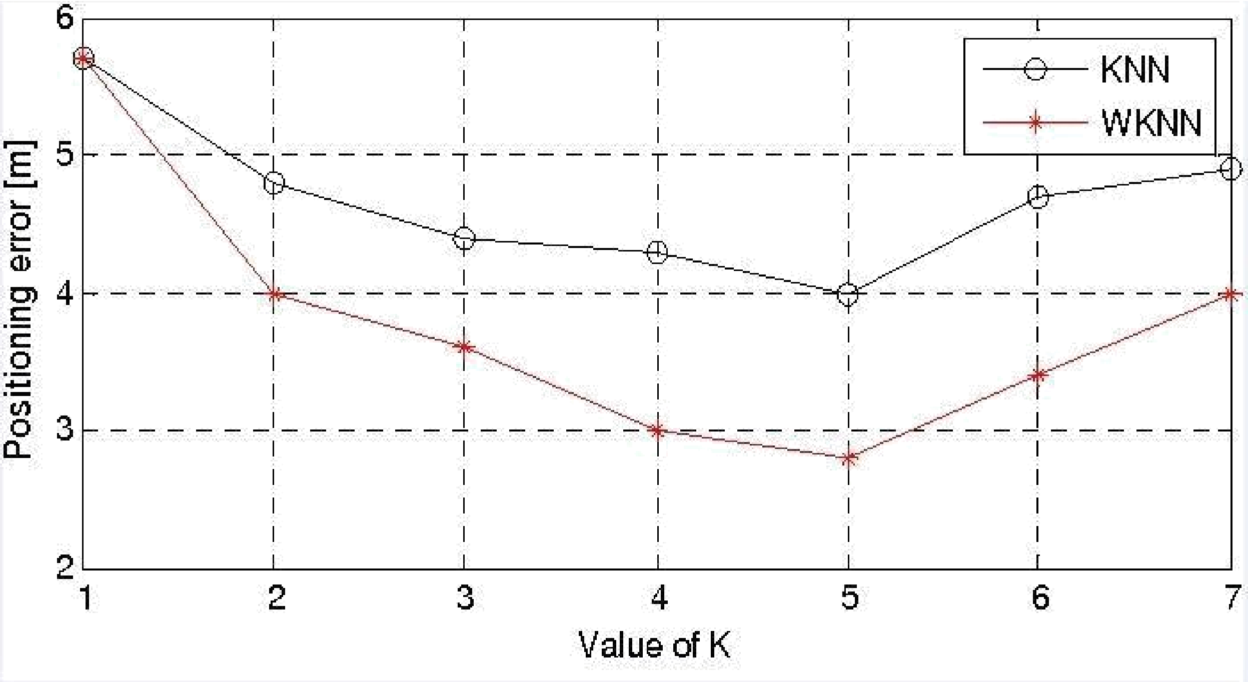


Graph 3:DensityClust

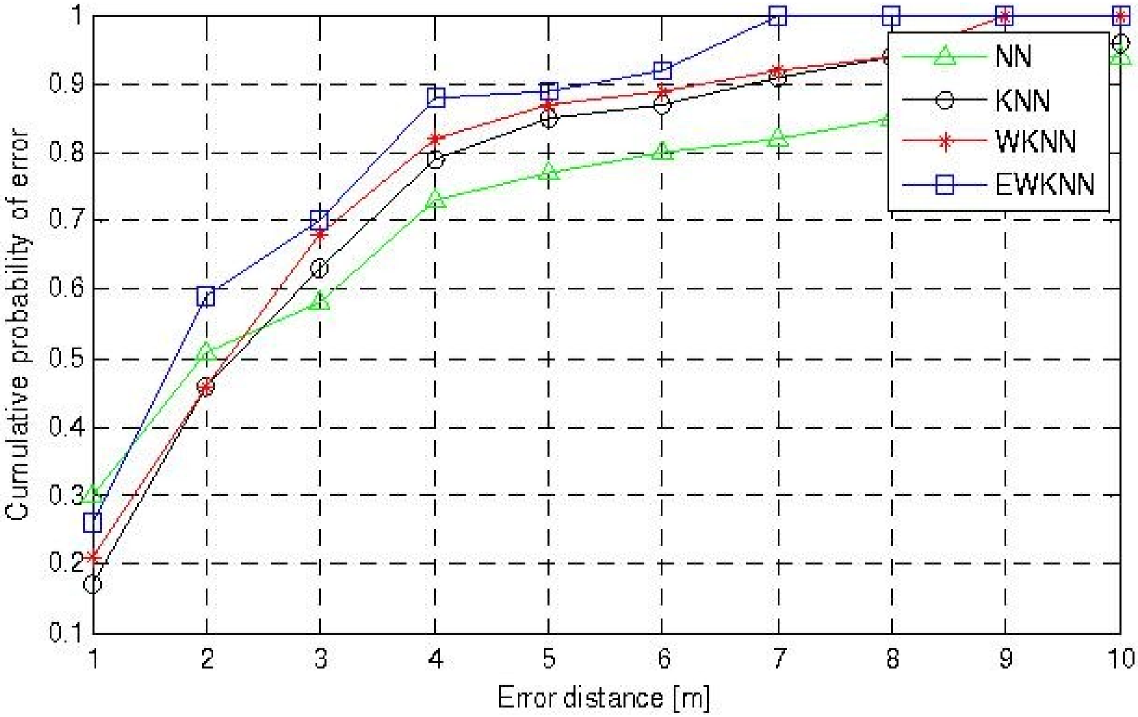
4.3 Comparison

K means	KNN
Unsupervised learning	Supervised learning
Use for Clustering	Use largely for Classification
K is the total clusters	K is the number of nearest neighbors
It is commonly used for market segmentation, anomaly detection, etc.	It is used for classification and regression of known data

Table 6:K means vs KNN method



Graph 4:K value



Graph 5: Error

Chapter-5 Conclusion

5.1 Conclusion:

K-Nearest neighbor classifier is a normal algorithm to learn classification based on variables and don't to develop a model from the given data set (train data set). However the classifying process put up very costly because it needs to calculates the same values one by one between the test and training examples. K-nearest neighbor classifier also affect from the various issue such as scaling . I calculates the closeness among the test example and training examples to perform classify the test data set. If the attributes have different values, the closeness of distance might be leads by one of the attributes, which is bad.

5.2 Future scope:

KNN classifier are applied to dimensionality reduced future value. The number of cases from the twenty patients to be increased better testing of the system

1. Features either the local or global used for recognition can be increased, to increase the efficiency of the object recognition
2. Geometric properties of the image can be included in the feature vector for recognition.
3. Using unsupervised classifier instead of a supervised classifier for recognition of the object.
4. The proposed object recognition system uses grey-scale image and discards the color information. The colour information in the image can be used for recognition of the object. Colour based object recognition plays vital role in Robotics.

5.3 Application of KNN

5.3.1 Nearest Neighbor:

Based Content Retrieval This is one the fascinating applications of KNN – Basically we can use it in Computer Vision for many cases – You can consider handwriting detection as a rudimentary nearest neighbor problem. The problem becomes more fascinating if the content is a video – given a video find the video closest to the query from the database – Although this looks abstract, it has lot of practical applications – Eg : Consider ASL (American Sign Language) . Here the communication is done using hand gestures. So lets say if we want to prepare a dictionary for ASL so that user can query it doing a gesture. Now the problem reduces to find the (possibly k) closest gesture(s) stored in the database and show to user. In its heart it is nothing but a KNN problem.

5.3.2 Gene expression:

Gene expression is the process by which information from a gene is used in the synthesis of a functional gene product. These products are often proteins, but in non-protein coding genes such as transfer RNA (tRNA) or small nuclear RNA (snRNA) genes, the product is a functional RNA. In genetics, gene expression is the most fundamental level at which the genotype gives rise to the phenotype, i.e. observable trait. The genetic code stored in DNA is "interpreted" by gene expression, and the properties of the expression give rise to the organism's phenotype. Such phenotypes are often expressed by the synthesis of proteins that control the organism's shape, or that act as enzymes catalysing specific metabolic pathways characterising the organism. Regulation is thus critical to an organism's development.

This is another cool area where many a time, KNN performs beer than other state of the art techniques. In fact a combination of KNN-SVM is one of the most popular techniques there. This is a huge topic on its own and hence I will refrain from talking much more about it.

The recent development of microarray technologies has enabled biologists to quantify gene expression of tens of thousands of genes in a single experiment. Biologists have begun collecting gene expression for a large number of samples. One of the urgent issues in the use of microarray data is to develop methods for characterizing samples based on their gene expression. The most basic step in the research direction is binary sample classification, which has been studied extensively over the past few years The process of building multiclass classifiers is divided into two components: (i) selection of the features (i.e. genes) to be used for training and testing and (ii) selection of the classification method

5.3.3 Research:

An object recognition system is developed, that recognizes the two-dimensional and three dimensional objects.

The feature extracted is sufficient for recognizing the object and marking the location of the object.

The proposed classifier is able to recognize the object in less computational cost.

The proposed global feature extraction requires less time, compared to the traditional feature extraction method.

The performance of the SVM-kNN is greater and promising when compared with the BPN and SVM.

The performance of the One-against-One classifier is efficient.

Global feature extracted from the local parts of the image.

Local feature PCA-SIFT is computed from the blobs detected by the Hessian-Laplace detector.

Along with the local features, the width and height of the object computed through projection method is used.

The methods presented for feature extraction and recognition are common and can be applied to any application that is relevant to object recognition.

The proposed object recognition method combines the state-of-art classifier SVM and k-NN to recognize the objects in the image. The multiclass SVM is used to hybridize with the k-NN for the recognition. The feature extraction method proposed in this research work is efficient and

provides unique information for the classifier.

5.4 References:

[1] *Usama Fayyad, Gregory Piatetsky-Shapiro, and Padhraic Smyth, "***From Data Mining to Knowledge Discovery in Databases",** Number 1996

[2] Shi Na, Liu Xumin, Guan yong, "**Research on k-means Clustering Algorithm",** 2010

[3] Yikun Qin, Zhu Liang Yu, Chang-Dong Wang, Zhenghui Gu,Yuanqing Li, "**A Novel Clustering Method based on Hybrid K-Nearest-Neighbor Graph**", November 2016

[4] Jianpeng Zhang, Mykola Pechenizkiy, Yulong Pei, Julia Efremova, "**A Robust Density-based Clustering Algorithm for Multi-Manifold Structure",** 2016

[5] Selahaddin Batuhan ,AkbenAhmet Alkan, "Density Weighted K-Nearest Neighbors Algorithm for Outliers in the Training Set Are So Close to the Test Element", November 2015

[6] Klaus Schliep & Klaus Hechenbichler, "**Weighted k-Nearest Neighbors**",April 2004

[7] Hamid Parvin,Hoseinali Alizadeh,Behrouz Minati, "**A Modification on K-Nearest Neighbor Classifier**", November 2010

[8] Hinneburg, A., Keim, "**An efficient approach to clustering in large multimedia databases with noise**." 1998, pp. 58–65.

[9] Jarvis, R.A., "**Clustering using a similarity measure based on shared nearest neighbors**" 1973.

[10] . Terrell, G.R., Scott, "**Variable kernel density estimation**", 1992.